Kementerian Pendidikan Dan Kebudayaan
Republik Indonesia
2013

KURIKULUM 2013

ADMINISTRASI SERVER



ADMINIDTRASI SERVER

1

UNTUK SMK/MAK X



Halaman Francis

Penulis : **Antonius Duty Susilo**

Editor Materi : **Wismanu**

Editor Bahasa :

Ilustrasi Sampul :

Desain & Ilustrasi Buku : **PPPPTK BOE MALANG**

Pusat Pengembangan & Pemberdayaan Pendidik & Tenaga Kependidikan Bidang Otomotif & Elektronika:

Jl. Teluk Mandar, Arjosari Tromol Pos 5, Malang 65102, Telp. (0341) 491239, (0341) 495849, Fax. (0341) 491342, Surel: vedcmalang@vedcmalang.or.id, Laman: www.vedcmalang.com



DISKLAIMER (DISCLAIMER)

Penerbit tidak menjamin kebenaran dan keakuratan isi/informasi yang tertulis di dalam buku tek ini. Kebenaran dan keakuratan isi/informasi merupakan tanggung jawab dan wewenang dari penulis.

Penerbit tidak bertanggung jawab dan tidak melayani terhadap semua komentar apapun yang ada didalam buku teks ini. Setiap komentar yang tercantum untuk tujuan perbaikan isi adalah tanggung jawab dari masing-masing penulis.

Setiap kutipan yang ada di dalam buku teks akan dicantumkan sumbernya dan penerbit tidak bertanggung jawab terhadap isi dari kutipan tersebut. Kebenaran keakuratan isi kutipan tetap menjadi tanggung jawab dan hak diberikan pada penulis dan pemilik asli. Penulis bertanggung jawab penuh terhadap setiap perawatan (perbaikan) dalam menyusun informasi dan bahan dalam buku teks ini.

Penerbit tidak bertanggung jawab atas kerugian, kerusakan atau ketidaknyamanan yang disebabkan sebagai akibat dari ketidakjelasan, ketidaktepatan atau kesalahan didalam menyusun makna kalimat didalam buku teks ini.

Kewenangan Penerbit hanya sebatas memindahkan atau menerbitkan mempublikasi, mencetak, memegang dan memproses data sesuai dengan undang-undang yang berkaitan dengan perlindungan data.

Katalog Dalam Terbitan (KDT)
Teknik Komputer Jaringan, Edisi Pertama 2013
Kementerian Pendidikan & Kebudayaan
Direktorat Jenderal Peningkatan Mutu Pendidik & Tenaga Kependidikan, th. 2013: Jakarta

KATA PENGANTAR

Puji syukur kami panjatkan kepada Tuhan yang Maha Esa atas tersusunnya buku teks ini, dengan harapan dapat digunakan sebagai buku teks untuk siswa Sekolah Menengah Kejuruan (SMK) Bidang Studi Administrasi Server

Penerapan kurikulum 2013 mengacu pada paradigma belajar kurikulum abad 21 menyebabkan terjadinya perubahan, yakni dari pengajaran (*teaching*) menjadi BELAJAR (*learning*), dari pembelajaran yang berpusat kepada guru (*teachers-centered*) menjadi pembelajaran yang berpusat kepada peserta didik (*student-centered*), dari pembelajaran pasif (*pasive learning*) ke cara belajar peserta didik aktif (*active learning-CBSA*) atau *Student Active Learning-SAL*.

Buku teks ″ Administrasi Server ″ ini disusun berdasarkan tuntutan paradigma pengajaran dan pembelajaran kurikulum 2013 diselaraskan berdasarkan pendekatan model pembelajaran yang sesuai dengan kebutuhan belajar kurikulum abad 21, yaitu pendekatan model pembelajaran berbasis peningkatan keterampilan proses sains.

Penyajian buku teks untuk Mata Pelajaran ″ Administrasi Server ″ ini disusun dengan tujuan agar supaya peserta didik dapat melakukan proses pencarian pengetahuan berkenaan dengan materi pelajaran melalui berbagai aktivitas proses sains sebagaimana dilakukan oleh para ilmuwan dalam melakukan eksperimen ilmiah (penerapan scientifik), dengan demikian peserta didik diarahkan untuk menemukan sendiri berbagai fakta, membangun konsep, dan nilai-nilai baru secara mandiri.

Kementerian Pendidikan dan Kebudayaan, Direktorat Pembinaan Sekolah Menengah Kejuruan, dan Direktorat Jenderal Peningkatan Mutu Pendidik dan Tenaga Kependidikan menyampaikan terima kasih, sekaligus saran kritik demi kesempurnaan buku teks ini dan penghargaan kepada semua pihak yang telah berperan serta dalam membantu terselesaikannya buku teks siswa untuk Mata Pelajaran Administrasi Server kelas X/Semester 1 Sekolah Menengah Kejuruan (SMK).

Jakarta, 12 Desember 2013

Menteri Pendidikan dan Kebudayaan

Prof. Dr. Mohammad Nuh, D

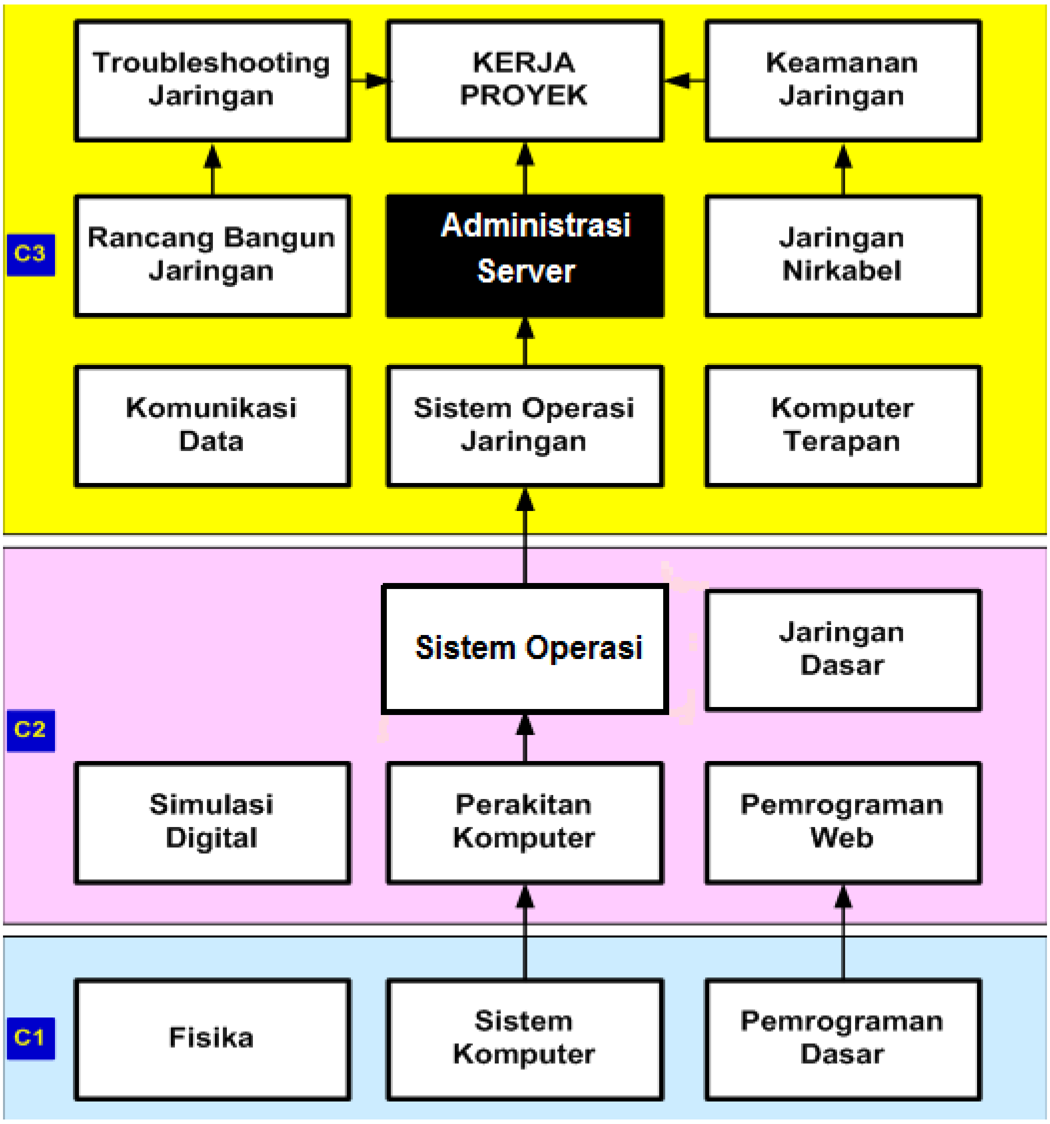


DAFTAR ISI


oo.tol
olo

PETA KEDUDUKAN BAHAN AJAR

Peta kedudukan bahan ajar merupakan suatu diagram yang menjelaskan struktur mata pelajaran dan keterkaitan antar mata pelajaran dalam satu kelompok bidang studi keahlian. Pada Gambar 0.1 dibawah ini, ditunjukkan peta kedudukan bahan ajar untuk program studi keahlian Teknik Komputer dan Jaringan (TKJ).



Gambar 0.1. Peta Kedudukan Bahan Ajar



BAB I PENDAHULUAN

A. Deskripsi

Administrasi berasal dari bahasa latin yaitu “Ad” dan “ministrate” yang artinya pemberian jasa atau bantuan, yang dalam bahasa Inggris disebut “Administration” artinya “To Serve”, yaitu melayani dengan sebaik-baiknya.

Pengertian administrasi dapat dibedakan menjadi 2 pengertian yaitu :

Administrasi dalam arti sempit. Menurut Soewarno Handayaningrat mengatakan “Administrasi secara sempit berasal dari kata Administratie (bahasa Belanda) yaitu meliputi kegiatan cata-mencatat, surat-menyurat, pembukuan ringan, keti-mengetik, agenda dan sebagainya yang bersifat teknis ketatausahaan”(1988:2). Dari definisi tersebut dapat disimpulkan administrasi dalam arti sempit merupakan kegiatan ketatausahaan yang mliputi kegiatan cata-mencatat, surat-menyurat, pembukuan dan pengarsipan surat serta hal-hal lainnya yang dimaksudkan untuk menyediakan informasi serta mempermudah memperoleh informasi kembali jika dibutuhkan.

Administrasi dalam arti luas. Menurut The Liang Gie mengatakan “Administrasi secara luas adalah serangkaian kegiatan yang dilakukan oleh sekelompok orang dalam suatu kerjasama untuk mencapai tujuan tertentu”(1980:9). Administrasi secara luas dapat disimpulkan pada dasarnya semua mengandung unsur pokok yang sama yaitu adanya kegiatan tertentu, adanya manusia yang melakukan kerjasama serta mencapai tujuan yang telah ditentukan sebelumnya.

Pendapat lain mengenai administrasi dikemukan oleh Sondang P. Siagian mengemukakan “Administrasi adalah keseluruhan proses kerjasama antara 2 orang atau lebih yang didasarkan atas rasionalitas tertentu untuk mencapai tujuan yang telah ditentukan sebelumnya” (1994:3). Berdasarkan uraian dan definisi tersebut maka dapat diambil kesimpulan bahwa administrasi adalah seluruh kegiatan yang dilakukan melalui kerjasama dalam suatu organisasi berdasarkan rencana yang telah ditetapkan untuk mencapai tujuan.

B. Prasyarat

Berdasarkan peta kedudukan bahan ajar pada Gambar 0.1. di atas, maka mata pelajaran sistem operasi ini mempunyai keterkaitan dengan mata pelajaran sistem komputer dan perakitan komputer. Sistem operasi merupakan perangkat lunak yang akan mengelola pemakaian perangkat keras atau sumber daya komputer. Untuk memahami pengelolaan yang dilakukan sistem operasi dibutuhkan pemahaman terhadap perangkat keras komputer baik secara *logical* dan *physical,* dimana topik ini telah diuraikan dalam mata pelajaran sistem komputer yang telah dipelajari pada kelas X semester I.

Untuk dapat mengoperasikan dan menggunakan komputer dengan baik, maka diperlukan satu set sistem komputer yang berfungsi dengan baik. Tahapan untuk menyiapkan bagaimana seperangkat sistem komputer dapat berjalan dengan baik, telah diuraikan dalam mata pelajaran perakitan komputer yang telah dipelajari pada kelas X semester

C. Petunjuk Penggunaan Buku Ajar

Buku Ajar ini disusun berdasarkan kurikulum 2013 yang mempunyai ciri khas menggunakan metode scientifik. Buku ini terdiri dari dua bab yaitu bab 1 pendahuluan dan bab 2 pembelajaran. Dalam bab pendahuluan beberapa yang harus dipelajari peserta didik adalah diskripsi mata pelajaran yang berisi informasi umum, rasionalisasi dan penggunaan metode scientifik. Selanjutnya pengetahuan tentang persyaratan, tujuan yang diharapkan, kompetensi inti dan dasar yang akan dicapai serta test kemampuan awal.

Bab 2 menuntun peserta didik untuk memahami diskrisi umum tentang topik yang akan dipelajari dan rincian kegiatan belajar sesuai dengan kompetensi dan tujuan yang akan dicapai. Setiap kegiatan belajar terdiri dari tujuan dan uraian materi topik pembelajaran, tugas serta test formatif. Uraian pembelajaran berisi tentang



BAB II PEMBELAJARAN

1. **Kegiatan Belajar 1 : Analisa Kebutuhan Server Untuk Lalu Lintas dan Aplikasi Jaringan Komputer**

a. Tujuan Pembelajaran

Setelah mengikuti kegiatan belajar 1 ini, siswa diharapkan dapat:

1. Memahami kebutuhan server untuk lalu lintas dan aplikasi jaringan komputer.

b. Uraian Materi

1. **Kebutuhan Server Untuk Lalu Lintas dan Aplikasi Jaringan Komputer**

   1. **Analisa Kebutuhan**

      Server adalah perangkat yang ditujukan untuk menyediakan sebuah layanan kepada beberapa perangkat lainya. Beberapa server bekerja dengan berat, dan beberapa lainya bekerja dengan ringan. Tidak semua kebutuhan server harus dimaksimalkan sebisa mungkin, beberapa kebutuhan bisa diminimalisir untuk menghemat biaya. Lalu, apa saja yang harus diperkirakan untuk mendapatkan server yang optimal?

      1. **Hardware**

         Sebelum memulai dengan spesifikasi bisnis, maka kita butuh untuk menentukan hardwarenya terlebih dahulu. Hardware yang dimaksud adalah hardware yang bekerja sama dengan server, seperti Router, Switch, dsb.

         Pastikan, bahwa server kita bisa bekerja sama dengan hardware lainya, pastikan bahwa port untuk koneksi antara peripheral satu bisa bekerja dengan yang lainya.

      2. **Aktifitas Bisnis**



Setelah hardware yang bisa bekerja sama dengan baik, maka kita harus tahu aktifitas bisnis kita seperti apa. Dalam analisa kedua ini, kita harus tahu seberapa berat aktifitas yang dilakukan oleh server, apakah server akan melakukan komputasi matematika yang rumit? Apakah server hanya menyediakan konten? Apakah server bekerja dengan keras sebagai penyedia jasa kriptografi?

Setelah mengetahui aktifitas bisnis, kita bisa menentukan kebutuhan hardware operasional dalam server kita, berapa clock rate yang dibutuhkan, jumlah RAM, jumlah penyimpanan data, dsb.

3. **Jumlah Pengguna**

   Selain aktifitas bisnis, maka yang tidak bisa lepas darinya adalah jumlah pengguna. Aktifitas bisnis yang ringan mungkin memerlukan kualitas hardware yang rendah, tapi dengan jumlah pengguna yang banyak, maka kualitas hardware yang tinggi juga dibutuhkan.

   $$\boldsymbol{optimal} = (\boldsymbol{n}_{\boldsymbol{pengguna}} * \boldsymbol{r}_{\boldsymbol{aktifitas\ bisnis}}) * \boldsymbol{n}_{\boldsymbol{pengguna}}$$

   Untuk mendapatkan spesifikasi yang minimal, jumlah pengguna dikalikan dengan rasio aktifitas bisnis yang dibutuhkan karena mereka saring terkait satu sama lain. Spesifikasi minimal tersebut dikalikan dengan jumlah pengguna lagi untuk membuat margin antar spesifikasi minimal dan rata-rata operasional.

4. **Skalabilitas**

   Sebuah struktur yang baik adalah apabila struktur tersebut tumbuh, dia tidak perlu dibangun lagi dari awal. Server harus bisa tumbuh dan berkembang sesuai dengan kebutuhan di masa mendatang. Penentuan skema jaringan, dan juga pembagian sumber daya yang tepat, alokasi cadangan yang tepat bisa membuat perkembangan server dengan mudah tanpa harus merubah yang sudah jadi.



5. **Titik Penyebaran**

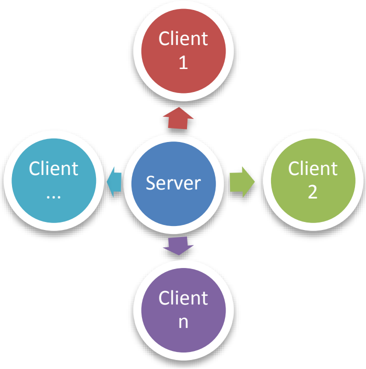
   Server dengan pengguna yang tersebar mempunyai jarak yang berbeda-beda dalam mengirimkan informasi. Apabila titik penyebaran terlalu jauh, bukan tidak mungkin server sekunder yang lebih dekat harus dibuat. Pembuatan server sekunder, bisa membuat penyampaian informasi, apabila tidak dimungkinkan, maka optimalisasi hardware media pengiriman data harus ditingkatkan, seperti menggunakan media kecepatan tinggi seperti kabel fiber, dsb.

6. **Software**

   Ingat bahwa hardware tidak bisa bekerja tanpa software. Pemilihan software yang tepat juga dibutuhkan. Kita juga bisa melihat aktifitas bisnis yang dilakukan, apakah server berfungsi sebagai penyedia DNS? Berarti, kita harus menginstall software manajemen DNS di server.

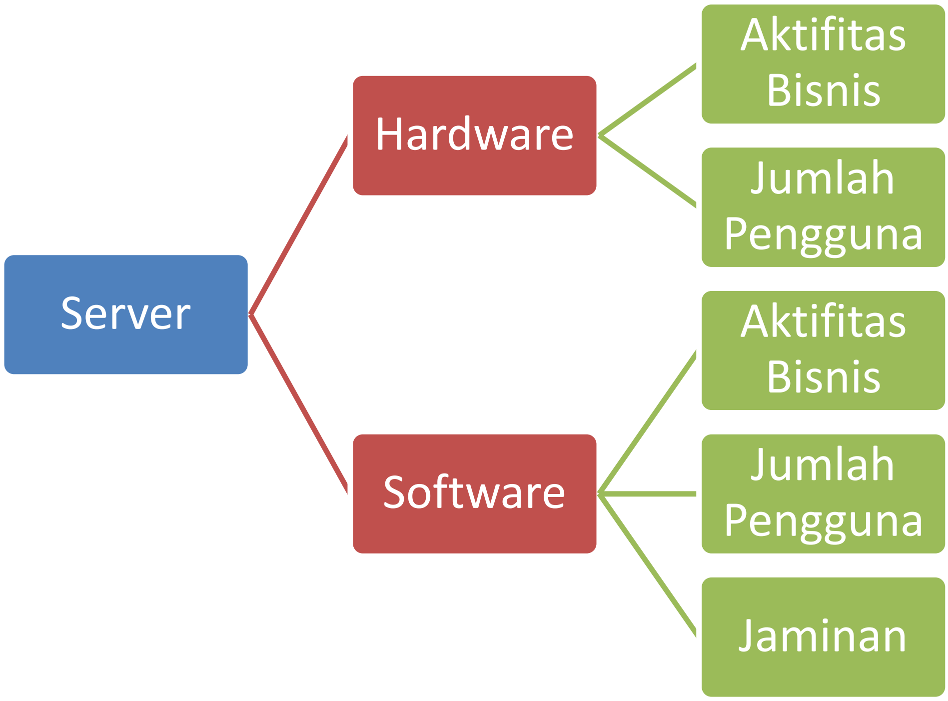
   Bagaimana dengan pemilihan jenis atau merk DNS server? Pemilihan yang sesuai tergantung dari kebutuhan server, apabila server melayani jutaan pelanggan tiap harinya dan apabila mati sejenak bisa menyebabkan bencana, maka pemilihan merk software sebaiknya dilakukan sebaik mungkin untuk meminimalisir kerugian.

   Server harus bisa dihandalkan untuk mengatasi jutaan pengguna yang mengaksesnya.





Secara hirarki, kebutuhan server terpisah antar hardware dan software, tapi salah satu dari mereka tidak bisa diabaikan. Hadware ditentukan berdasarkan aktifitas bisnis dan jumlah pengguna, lalu bagaimana untuk menentukan hardware ini? Ada beberapa metode, salah satunya adalah klasifikasi tingkatan. Klasifikasi ini bisa berbeda-beda, sesuai dengan orang yang melakukan klasifikasi, orang tersebut harus berpengalaman dalam melakukan perkiraan kebutuhan hardware untuk sebuah bisnis.



Klasifikasi Tingkatan

Klasifikasi tingkatan dilakukan dengan membagi jenis-jenis hardware, seperti kebutuhan RAM, Hardisk, CPU Clock Rate, dll.

Sebelum melakukan optimalisasi, maka formula perhitungan optimalisasi di atas harus dinormalisasikan.

Contoh kasus, perusahaan dengan pegawai sebanyak 1000 orang, dengan aktifitas bisnis melakukan pengunggahan laporan, pengunduhan laporan, dan validasi data. Dilakukan benchmark, bahwa dalam 1 menit, ada 500 orang melakukan aktifitas bisnis, dari aktifitas tersebut, 30 orang gagal, dan sisanya

berhasil. Maka rasio aktifitas bisnisnya adalah 500-30/500, 94%, ada 6% yang gagal, sehingg rasionya adalah 106%.

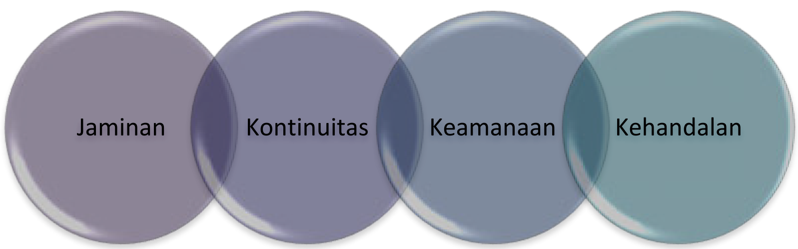
Maka angka optimalisasinya, adalah 1000 * 1.0 * 1000 = 1,000,000.
Rasio akfitifas bisnis di optimalisasi harus 100%.
Lalu, dilakukan pencarian aktifitas puncak, 1 * 1.6 * 1000 = 1.600.000.
Rasio optimalisasinya 1.000.000/1.600.000 = 0.625

Tabel Klasifikasi RAM

Ukuran RAM (GB)	Rasio Optimalisasi
2	< 10%
4	> 10%
8	> 40%
16	> 50%

Maka RAM yang dibutuhkan adalah 16 Giga Byte.
Begitu pula dengan cara perhitungan jumlah memory, clock rate CPU, dsb. Harus diingat bahwa tabel klasifikasi ini harus ditentukan oleh orang yang benar-benar mengerti tentang optimalisasi, yang telah berpengalaman dalam melakukan klasifikasi kebutuhan.
Tabel klasifikasi muncul seiring dengan orang yang bertanggung jawab untuk melakuka klasifikasi tersebut mempunyai pengalaman saat operasional lapangan berlangsung.

Pemilihan Merk Software



Ada banyak sekali merk web server di luar sana. Lalu, bagaimana kita bisa memilih merk yang sesuai? Untuk mendapatkan kemampuan server yang optimal, maka merk dari web server harus memenuhi 4 kriteria berikut.

1. **Jaminan**

   Aplikasi web server harus terjamin, dan apabila terjadi kerugian maka perusahaan web server tersebut juga harus ikut mempertanggung-jawabkanya.

2. **Kontinuitas**

   Jaman berkembang terus, merk web server harus terus dikembangkan seiring dengan berkembangnya teknologi, dalam kasus web server seperti versi protocol HTTP, web server harus terus berkembang dan mengikuti standar prototokl HTTP yang baru tanpa melupakan yang lama.

3. **Keamanan**

   Web server yang aman, tidak rentan terhadap pencurian identitas, injeksi, dsb, adalah salah satu alasan kemampuanya untuk terus berjalan dan melayani pengguna.

4. **Kehandalan**

   Web server harus handal, dia tidak boleh sangat lemah hingga bahkan apabila hardware server sudah optimal, tapi software web server hanya bisa melayani 256 pengguna setiap menit.

c. Rangkuman

Server adalah perangkat jaringan yang memberikan layanan terhadap pengguna. Server melayani pengguna dengan handal dan harus bisa melakukanya setiap saat. Maka analisa terhadap kebutuhan server untuk bisa memenuhi tugasnya sangat dibutuhkan.

Server yang lemah, tidak sesuai, dan tidak handal, bisa membuat kerugian yang besar. Oleh karena itu, ada beberapa faktor yang harus dipertimbangkan sebelum membuat sebuah server.



Server harus disesuaikan berdasarkan kebutuhan kesesuaian dengan hardware jaringan yang lainya, aktifitas bisnis yang dilakukanya, jumlah pengguna yang dilayaninya, juga harus dilengkapi dengan software yang mendukung kinerja hardware secara optimal, handal, dan terjamin

d. Tugas

Jawab pertanyaaan berikut ini :

1. Apa saja yang harus diperkirakan untuk mendapatkan server yang optimal?
2. Bagaimana ciri hardware yang baik untuk digunakan di server?
3. Bagaimana jumah pengguna bisa mempengaruhi kinerja hardware?
4. Bagaimana caranya agar server bisa terus berkembang?
5. Bagaimana bisa ukuran RAM mempengaruhi kinerja server? Jelaskan!
6. Bagaimana *bottleneck* bisa terjadi, dan dalam kasus apa?
7. Sebutkan dan jelaskan 4 faktor utama pemilihan software untuk server!
8. Secara kesatuan, apakah aktifitas bisnis, jumlah pengguna berpengaruh terhadap kecepatan pelayanan server? Bagaimana bisa? Jelaskan!
9. Apakah cukup membuat server dengan software yang bagus tapi hardware pas-pasan? Jelaskan!
10. Apa saja yang bisa dilakukan untuk membuat server bekerja dengan optimal?

e. Test Formatif

1) Apakah yang dimaksud dengan server?

   a) Perangkat yang mengirim informasi
   b) Perangkat untuk berbagi file
   c) Perangkat yang melayani pengguna
   d) Perangkat pengaman sistem
   e) Perangkat pelengkap sistem

2) Pemilihan hardware sangat penting dan menentukan kehandalan server, kenapa?
   a) Hardware merupakan komponen penyusun server
   b) Hardware merupakan komponen pembantu server
   c) Hardware merupakan kompenen operasi server
   d) Hardware merupakan komponen hidup server
   e) Semua benar
3) Aktifitas bisnis yang sangat berat contohnya adalah ...
   a) Penyedia kriptografi
   b) Penyedia konten file HTML statis
   c) Penyedia aktifitas transaksi koperasi
   d) Penyedia cloud
   e) Penyedia penyimpanan file
4) Jumlah pengguna yang banyak harus didukung dengan hardware yang memadai, dalam hal apa biasanya jumlah pengguna bisa banyak?
   a) Operasi transaksi bank dunia
   b) Operasi transaksi jual beli toko lokal
   c) Operasi penyedia game online lokal
   d) Operasi pemetaan lokasi militer
   e) Semua salah
5) Cara yang tepat apabila server hanya satu sedangkan pengguna tersebar jauh adalah ...
   a) Membuat jarak pengguna semakin dekat
   b) Mengoptimalkan data untuk pengiriman jarak jauh
   c) Memprioritaskan pengguna jauh
   d) Meningkatkan media pengiriman ke pengguna
   e) Semua benar
6) Hardware berimbas langsung terhadap kinerja optimal server, karena hardware merupakan ...
   a) Sistem utama, tempat operasi yang diberikan software terjadi di sini
   b) Sistem pembantu, apabila software melemah, maka hardware yang membantu



c) Sistem cadangan, apabila software mati, maka hardware yang bekerja

d) Semua benar

e) Semua salah

7) Yang bukan salah satu faktor yang mempengaruhi pemilihan software adalah ...

a) Jaminan

b) Kontinuitas

c) Keamanan

d) Keseragaman

e) Kehandalan

8) Yang dimaksud dengan kontinuitas adalah ...

a) Software terus berkembang mengikuti teknologi yang ada

b) Software bisa berjalan terus dengan lancar

c) Software mengirimkan data terus-menerus

d) Software melayani pengguna terus-menerus

e) Semua salah

9) Penggunaan lebih dari satu server sangat berpengaruh karena ...

a) Konten yang diberikan bisa lebih banyak

b) Jarak pengguna satu dengan server yang lain mungkin lebih dekat sehingga waktu interaksi bisa lebih cepat

c) Beban tiap server bisa dikurangi

d) Jawaban B dan C benar

e) Jawaban A dan C benar

10) Apabila konten yang ada di server berjumlah jutaan, maka ...

a) Server akan lambat karena konten yang ada banyak

b) Server akan mati karena konten terlalu banyak

c) Tidak ada pengaruh, karena konten hanya disimpan

d) Berpengaruh, karena konten mempengaruh proses server

e) Semua salah



f. Lembar Jawaban Tes Formatif

g. Lembar Kerja Siswa

2. Kegiatan Belajar 2 : Menyajikan Laporan Hasil Analisa Kebutuhan Server Untuk Lalu Lintas dan Aplikasi Jaringan Komputer

a. **Tujuan Pembelajaran**

Setelah mengikuti kegiatan belajar 2 ini, siswa diharapkan dapat:

1. Menyajikan laporan hasil analisa kebutuhan server untuk lalu lintas dan aplikasi jaringan komputer.

b. **Uraian Materi**

2. **Penyajian Laporan Analisa Kebutuhan Server Untuk Lalu Lintas dan Aplikasi Jaringan Komputer.**

1. Penyajian Laporan

Laporan dari hasil analisa kebutuhan server harus disajikan dengan jelas dan rinci, serta tidak terlalu membingungkan. Laporan harus padat, dan tidak boleh ada kesalahan penulisan, terutama di bagian angka, karena angka kebutuhan server sangat krusial.

Format penyajian laporan tidak baku, dan sesuai dengan kebutuhan yang diminta perusahaan atau bahkan membuatnya sendiri. Namun, informasi yang dikoleksi harus jelas.

Informasi-informasi penting yang harus dikoleksi saat membuat laporan adalah.

1. Kemampuan minimal server.
   i. Kemampuan minimal server, meliputi kapasitas RAM, Hardisk, Clock Rate CPU, Perangkat IO, Up Time, Power.
   ii. Semua itu diukur saat server berada di posisi paling buruk, dengan kemampuan ini bisa diketahui angka operasional kemampuan operasi server.



2. Kemampuan optimal server.
   i. Kemampuan optimal server, meliputi kapasitas RAM, Hardisk, Clock Rate CPU, Perangkat IO, Up Time, Power.
   ii. Semua itu diukur saat server berada di posisi normal, dengan kemampuan ini bisa diketahui angka operasional kemampuan operasi server.
   iii. Software atau layanan yang didukung oleh server, seperti Web, FTP, DNS, DHCP, dsb.

   Biasanya, hanya kemampuan optimal saja yang dibutuhkan. Cara penyajian laporan bisa berbentuk tabel yang mudah dibaca.

Hardware	Spesifikasi	Biaya
RAM	16GB	Rp1.000.000
Hardisk	16TB	Rp8.000.000
CPU Clock	5.6GHz	Rp3.000.000
Perangkat IO	Non Blocking, 1TBps	Rp1.000.000
Up Time	24x7x4x52	-
Power	440v	Rp500.000

Kemampuan Optimal

Software	Merk	Biaya
Web Server	Apache 2.0	-
Database Server	PostgreSQL	-
FTP Server	GloriaFTP	-

Jumlah Pengguna : 1500/hari.
Aktifitas : Unggah data, Unduh data, validasi data.
Transaksi Data : 200 Transaksi/menit.

Beberapa tambahan, seperti merk dari hardware juga bisa disajikan, atau informasi lebih rinci tentang kemampuan software juga bisa disajikan.

c. **Rangkuman**

Penyajian laporan analisa kebutuhan server merupakan cara penyajian informasi yang sudah dikoleksi dengan cara uji coba kemampuan server, penggunaan tabel klasifikasi, sehingga bisa dibaca dengan mudah dan tepat sasaran.

Penyajian laporan tidak mempunyai format khusus, satu-satunya hal yang perlu diperhatikan adalah informasi yang tertera, spesifikasi hardware yang dibutuhkan, software yang dibutuhkan, juga hasil dari kemampuan tersebut.

d. **Tugas**

1. Buatlah sebuah spesifikasi sistem yang cocok untuk server yang melayani web server social media dengan konten berbagi foto, dengan aktifitas pengguna 400,000/jam.
2. Buatlah laporan tentang cara memenuhi spesifikasi untuk mencapai kriteria di tugas nomor satu.

e. **Tes Formatif**

f. **Lembar Jawaban Tes Formatif**

g. **Lembar Kerja Siswa**

3. Kegiatan Belajar 3 : Memahami Tugas dan Tanggung Jawab Admin Server

a. Tujuan Pembelajaran :

Setelah mengikuti kegiatan belajar 3 ini, siswa diharapkan dapat:

3. Memahami tugas dan tanggung jawab admin server.
4. Menerapkan

b. Uraian Materi

5. Memahami Tugas dan Tanggung Jawab Admin Server

1. Tugas dan Tanggung Jawab Admin Server

Bekerja sebagai admin server merupakan sebuah pekerjaan yang menantang, tidak hanya bertanggung jawab untuk membuat server, admin server dituntut untuk merawat server, sehingga bisa hidup dan berkembang seterusnya sesuai kebutuhan.

Admin server mempunyai berbagai tugas dan peranan penting di dalam pengelolaan jaringan. Tugas admin server, berbeda di antara perusahaan satu dengan yang lainya. Tugas utamanya, adalah membangun, mengelola server, dan memperbaiki server.



Admin server bekerja sebaik mungkin untuk membuat server berjalan dengan lancar, melakukan perbaikian terjadwal, memastikan keamanan server, juga membantu pekerja lainya untuk menjaga keadaan server tetap optimal.

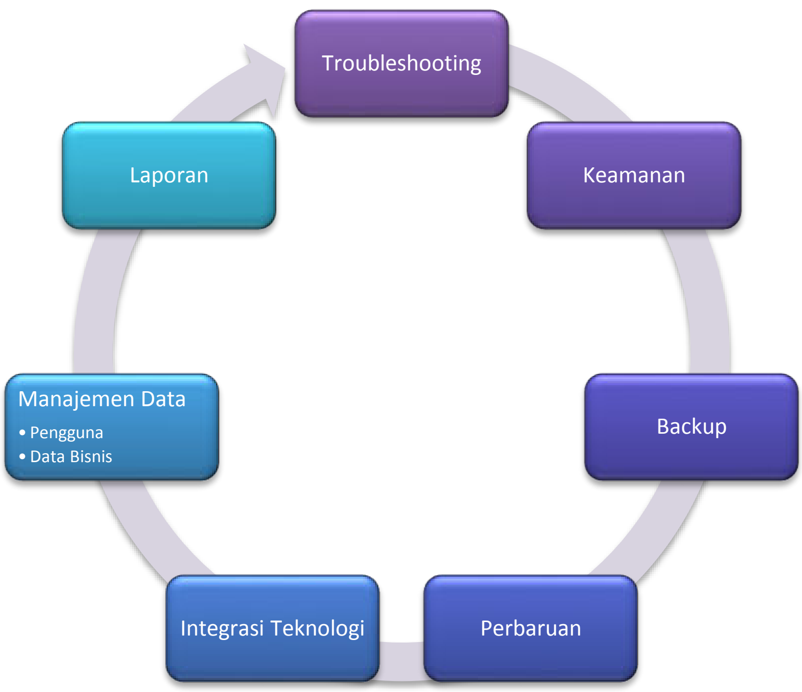


Tidak hanya bekerja dengan mesin, admin server juga harus bekerja sama dengan pekerja lainya, supervisor, hingga pekerja teknik. Melakukan pemecahan masalah yang sedang terjadi, juga memberikian pelayanan pelanggan. Sehingga admin server juga harus bisa menjelaskan berbagai hal teknis kepada orang awam.

Selain menjaga yang sudah ada, admin server juga harus menambahkan software atau melakukan update untuk terus membuat server bekerja dengan sempurna. Menambahkan pengguna, melakukan pembaharuan kata sandi, melakukan backup rutin, mengadaptasi teknologi baru, dan melakukan konfigurasi sesuai keadaan.

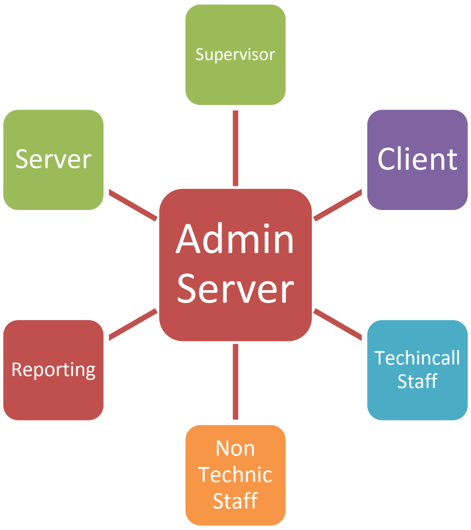
Apabila keadaan normal, maka biasanya ada rutin khusus di perusahaan tentang apa yang harus dilakukan admin server,

1. Mendatangi server, melihat aktifitas log, apakah ada yang mencurigakan.
2. Apabila ada kecurigaan, maka dilakukan troubleshooting.
3. Mengecek usia software yang digunakan, apakah ada update.
4. Apabila ada update, lakukan backup terlebih dahulu, baru lakukan update.
5. Melaukan integrasi dengan teknologi baru jika ada, dan melakukan konfigurasi untuk menyesuaikan keadaan perusahaan.
6. Mengatur pengguna, menambah pengguna baru atau menghapus



pengguna lama.

Sistem kerja admin server, yang biasanya dilakukan di tiap harinya. Ingat, admin server tidak hanya bekerja sendirian.



c. Rangkuman

Admin server adalah orang yang mengatur aktifitas server, mulai dari awal dibuat, perawatan, hingga perbaikan.

Admin server harus bisa menyesuaikan keadaan, melihat kemampuan server dan mengoptimalkanya. Melakukan perbaikan, mencari masalah dengan server untuk membuatnya lebih baik, melakukan perbaruan software dan mengkonfigurasi ulang untuk server.

Admin server tidak hanya bekerja sebagai petugas yang berhubungan dengan mesin, tapi juga harus bisa bekerja sama dengan pekerja lainya,

supervisor, atau bahkan pekerja non teknis, atau pelanggan yang membutuhkan layanan.

d. Tugas

1. Siapa yang dapat bertugas sebagai admin server?
2. Bagaimana cara admin server menyesuaikan dengan keadaan server yang rendah?
3. Siapa saja yang harus dihadapi oleh admin server?
4. Jelaskan cara admin server bekerja, dan apa yang dikerjakan secara singkat!
5. Apa yang harus dilakukan admin saat terjadi kegagalan pada sistem server.

e. Test Formatif

Pilih jawaban yang paling tepat!

1. Di bawah ini yang bukan merupakan tugas utama dari seorang admin server adalah ....
   a. membangun server
   b. mengelola server
   c. memperbaiki server
   d. memelihara server
   e. membuat laporan server
2. Yang harus dilakukan oleh seorang admin server adalah seperti di bawah ini, kecuali ...
   a. membuat server berjalan lancar
   b. melakukan perbaikan terjadwal
   c. melakukan pengecekan komputer client
   d. memastikan keamanan server
   e. membantu pekerja lainnya untuk menjaga keadaan server tetap optimal
3. Selain menjaga yang sudah ada pada server, seorang admin server juga harus ...
   a. mendatangi client, melihat aktifitas log
   b. membantu bagian pelayanan pelanggan
   c. melakukan update hardware setiap waktu



d. menambahkan atau melakukan update software
e. melakukan konfigurasi setiap saat

4. Berikut ini merupakan pekerjaan rutin khusus yang harus dilakukan oleh admin server di perusahaan, kecuali ...
   a. Mendatangi server, melihat aktifitas log, apakah ada yang mencurigakan
   b. Apabila ada kecurigaan, maka dilakukan troubleshooting
   c. Mengecek usia software yang digunakan, apakah ada update
   d. Apabila ada update, lakukan backup terlebih dahulu, baru lakukan update
   e. Melakukan konfigurasi setiap hari untuk menyesuaikan keadaan perusahaan
5. Selain masalah teknis dan mesin, admin server juga bisa bekerja dengan ...
   a. Supervisor
   b. Software
   c. Hardware
   d. jaringan internet
   e. jaringan intranet

f. Lembar Jawaban Tes Formatif

g. Lembar Kerja Siswa

4. Kegiatan Belajar 4 : Menalar Tugas dan Tanggung Jawab **Admin Server**

a. Tujuan Pembelajaran

Setelah mengikuti kegiatan belajar 4 ini, siswa diharapkan dapat:

6. Menalar tugas apa saja yang dibebankan kepada admin server, tertulis ataupun tidak.

b. Uraian Materi

7. **Penalaran Tugas dan Tanggung Jawab Admin Server**



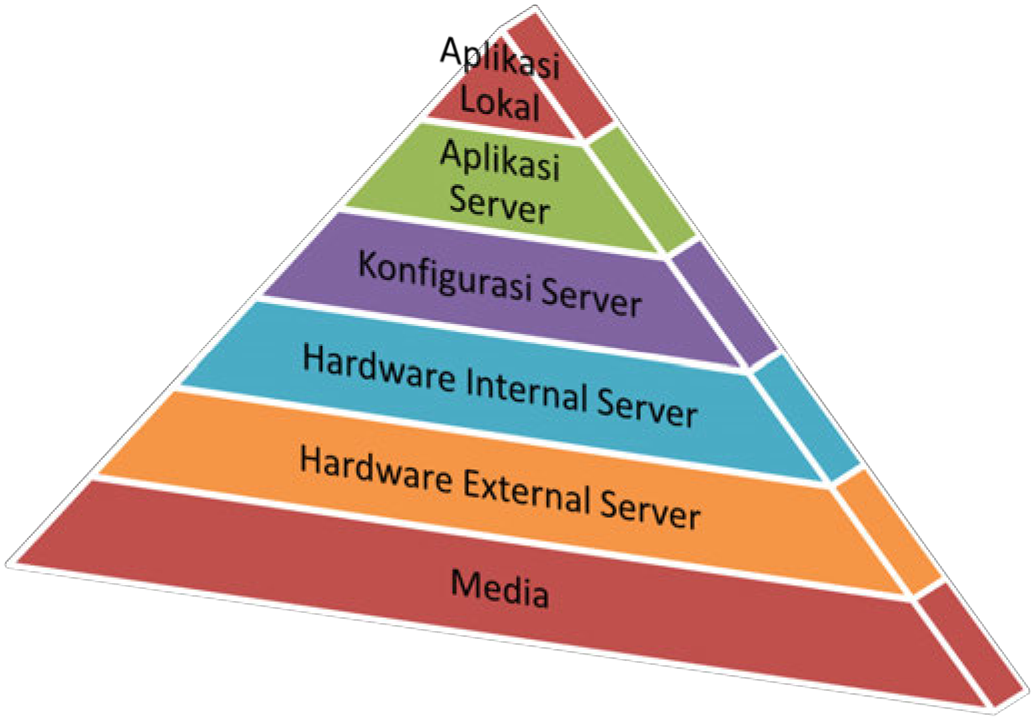
1. Menalar Tugas dan Tanggung Jawab Admin Server

Admin tidak hanya bisa melakukan pekerjaan prosedural, secara tetap dan statis. Kemungkinan yang terjadi di server berbeda-beda, sehingga admin server harus bisa menyesuaikan dengan keadaan.

Admin server harus bisa menalar, mengira-ngira apa yang harus dilakukan, bagaimana cara menanggulangi suatu masalah. Seperti, saat terjadi kesalahan pada server, atau terjadi serangan pada server. Admin server harus bisa mengantisipasi segala sesuatu yang sebelumnya tidak tertulis sebagai tugasnya.

Apakah tugas admin server selalu mudah seperti protokol yang sudah berlaku? Admin server bertanggung jawab untuk mengatasi segala masalah apapun caranya. Bagaimana cara memecahkan masalah server?

Ada beberapa tahapan yang harus dilakukan untuk memecahkan suatu masalah, pendekatan-pendekatan khusus untuk mencari akar permasalahan harus dilakukan.



Rangkaian masalah berurutan berbentuk piramid. Apabila yang bawah rusak, maka lapisan atas dipastikan tidak akan bisa berjalan. Lalu bagaimana pendekatan untuk melakukan *troubleshooting* atau pemecahan masalah.

Ada dua cara metode, dari atas ke bawah, atau dari bawah ke atas. Pendekatan pemecahan masalah dari bawah merupakan yang paling efektif, dilakukan pengecekan pada media penghubung server dan client. Apakah ada masalah? Lalu dilanjutkan mengecek hardware eksternal dari server, yaitu hardware yang berhubungan dengan media seperti NIC. Apakah ada masalah? Lalu dilanjutkan mengecek hardware internal server seperti hardisk corrupt, RAM, apakah ada yang rusak? Lalu dilanjutkan mengecek konfigurasi server, adakah konfigurasi yang salah diketikkan? Lalu dilanjutkan mengecek aplikasi server, seperti DNS, Web Server, adakah kesalahan pada aplikasi server? Lalu dilanjutkan pada aplikasi yang mengimplementasikan layanan dari aplikasi server, adakah kesalahan?



Pengecekan dari bawah ke atas bisa digunakan untuk memecahkan masalah secara cepat, karena apabila cuma media yang rusak, kita tidak perlu memecah dari atas.

Namun, melakukan pengecekan secara buta juga bisa membuat troubleshooting berjalan lama. Pengecekan yang paling akurat, dan cepat, adalah melakukan pencatatan atas setiap perubahan yang terjadi.

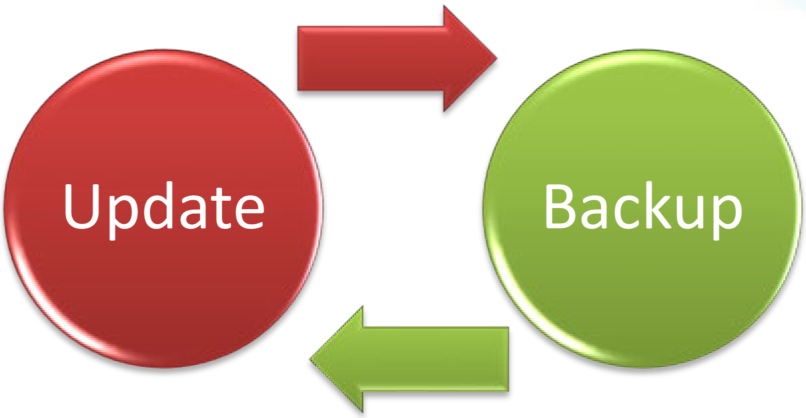
Subjek	Perubahan	Waktu
Media	Pengkabelan ulang RJ45 antara Server-1 dengan Router-1.	03 Oktober 2013
Konfigurasi Server	Pengaturan ulang NAT	04 Oktober 2013
Konfigurasi Aplikasi Server	Web Server Apache, Penambahan Host	05 Oktober

Dengan menuliskan log perubahan, maka kita bisa tahu kira-kira bagian mana yang terakhir kali dirubah, dan mengapa, dan kemungkinan terjadi kesalahan saat perubahan itu.

Admin server harus terus membuat servernya stabil, dengan performa optimal. Melakukan update dan *patch* di berbagai sistem server. Tidak serta-merta langsung melakukan update, admin server harus tahu apakah update tersebut diperlukan. Dia juga harus melakukan backup terlebih dahulu sebelum melakukan update untuk melindungi server dari kesalahan yang tidak diinginkan gara-gara update.

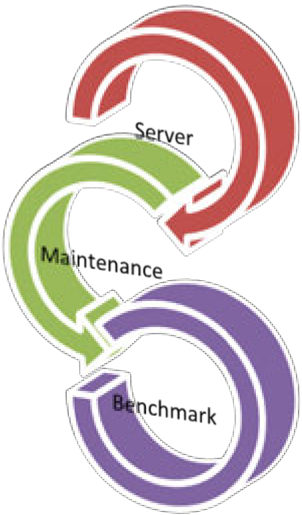
Dengan melakukan update dan patch, maka keamanan dan kehandalan server bisa terjamin.





Proses update dan backup selalu beriringan, tidak boleh tidak. Sebelum dilakukan update, maka terlebih dahulu lakukan backup.

Proses yang sama juga dilakukan ketika melakukan integrasi dengan teknologi baru. Admin server harus bisa melakukan benchmark, uji coba dan mendapatkan hasil statistik tentang operasional server. Lalu melakukan perbaikan untuk membuat hasil benchmark yang memuaskan.
Tidak hanya memperbaiki ketika rusak, sebisa mungkin admin server harus mengantisipasi kerusakan. Bagaimanapun caranya, kerusakan harus diminimalisir. Apabila ada beberapa bagian yang membuat server bekerja dengan lambat, mungkin karena salah konfigurasi, terlalu banyak file-file sampah dan juga data-data tidak berguna.



Harus dilakukan maintenance secara berkala terhadap server, dan lihat apa yang terjadi setelah maintenance dengan melakukan benchmark. Apabila setelah maintenance terjadi penurunan nilai benchmark, pasti ada sesuatu yang terjadi saat maintenance.

Hal-hal seperti ini harus diatasi oleh admin, untuk membuat server yang dimanajemen berjalan dengan optimal.

c. Rangkuman

Melakukan penalaran tugas admin server adalah cara untuk melaksanakan tugas yang tidak tertulis di tata cara atau tugas utama admin server.

Admin server dituntut untuk menyelesaikan masalah bagaimanapun itu, dengan cara apapun, yang penting server berjalan dengan lancar dan aman. Cara pemecahan bisa bermacam-macam, namun intinya adalah penelusuran dalam bentuk piramid.

Care bentuk piramid melakukan pengambilan informasi dari dasar, sehingga akar permasalahan bisa dipecahkan satu-persatu hingga ketemu titik dimana terjadi kesalahan dan bagaimana memperbaikinya.

Cara yang paling akurat dan cepat adalah dengan mencatat setiap perubahan yang terjadi pada sistem, selain pertanggung-jawaban yang jelas, penelusuran kesalahan bisa dilakukan secara tepat, dan tidak membuang waktu menelusuri permasalahan dari bawah.

Dengan berbagai macam cara, admin server harus membuat server berjalan dengan optimal dengan update, mengatur konfigurasi paling optimal, menerapkan teknologi baru, sehingga server terus berkembang dan bisa sesuai dengan keinginan kita.

Tidak hanya saat keadaan mendesak, sebisa mungkin admin server membuat keadaan mendesak menghilang dengan mengantisipasinya jauh hari dengan melakukan pengecekan rutin.

d. Tugas

Buat laporan lengkap untuk pertanyaan di bawah ini :

1. Apa saja tahap-tahap proses yang harus dilakukan oleh admin server jika terjadi serangan pada server?
2. Bagaimana prosedur untuk melakukan backup, restore dan update pada server?
3. Bagaimana prosedur untuk mencatat log perubahan pada server?

e. Tes Formatif

Jawab pertanyaan berikut ini :

1. Jelaskan tahapan yang harus dilakukan oleh admin server untuk memecahkan suatu masalah?
2. Pendekatan pemecahan masalah apakah yang merupakan langkah merupakan paling efektif bagi admin server?
3. Apa saja tugas dan tanggung jawab seorang admin server? Jelaskan secara singkat!
4. Jelaskan proses pemecahan masalah yang dilakukan dari bawah berdasarkan pendekatan rangkaian masalah berbentuk piramid!
5. Mengapa diperlukan penulisan log perubahan?
6. Apa fungsi dari proses backup dan update?
7. Bagaimana proses untuk meminimalisasi kerusakan pada server?
8. Jelaskan kemungkinan apa saja yang bisa membuat server bekerja lambat?
9. Apa saja yang harus dilakukan oleh admin server untuk memecahkan masalah hardware internal server?

10. Apa saja yang harus dilakukan oleh admin server untuk memecahkan masalah konfigurasi server?

f. Lembar Jawaban Tes Formatif

g. Lembar Kerja Siswa

5. Kegiatan Belajar 5 : Memahami Prinsip Kerja Komunikasi Client Server

a. Tujuan Pembelajaran

Setelah mengikuti kegiatan belajar 5 ini, siswa diharapkan dapat:

8. Memahami prinsip kerja komunikasi client server.

b. Uraian Materi

9. **Prinsip Kerja Komunikasi Client Server.**

1. Prinsip Kerja Komunikasi Client Server

Server tanpa client sama seperti rumah tanpa penghuni, maka dibutuhkan client yang menggunakan jasa dari server. Client dan server saling berhubungan secara timbal balik. Server secara selektif menyediakan sumber daya yang dibutuhkan, dan client melakukan koneksi ke server untuk meminta sumber daya tersebut.

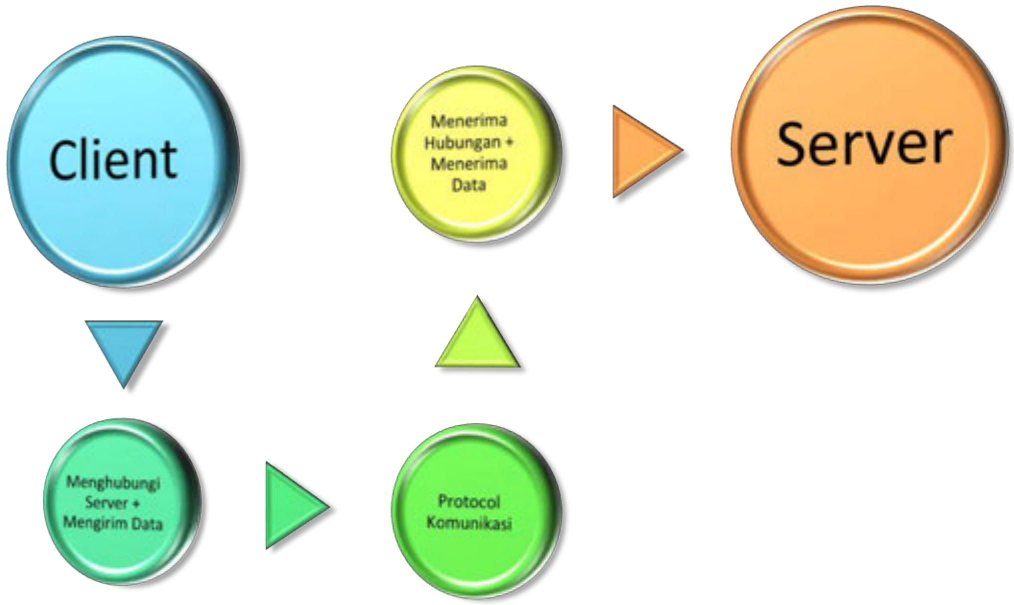
Komunikasi client server bekerja dengan cara *request-response*, dimana client meminta lalu server mengirim. Antara client dan server harus menggunakan aturan yang sama, kapan mengirim, kapan menerima, dan apa yang harus dikirim dan diterima. Semua aturan ini dinamakan *protocol*, yaitu cara komunikasi antara dua pihak atau lebih.

Server menyediakan berbagai macam layanan, web server menyediakan halaman web, file server menyedakan file computer. Tapi, secara tidak



langsung, untuk memproses halaman atau file yang diminta, server harus menggunakan sumber dayanya seperti RAM, Hardisk, CPU, dsb.

Bagaimana cara server meminta dari server? Dan bagaimana server memberi kepada client?



Client menghubungi server dan mengirim data yang dibutuhkan apabila ada, semua proses tersebut membutuhkan protocol. Protocol mengatur cara client/server mengirim data dan menerima data.

Sebagai contoh,

Ada sebuah bank yang memiliki **server**, nasabah menggunakan **web browser** (client) untuk mengakses data dari server. Nasabah meminta halaman web melalui **web server** dengan **protocol HTTP**. Server meminta nasabah untuk memasukkan informasi nama dan kata sandi untuk masuk ke dalam server.

Server menjalankan aplikasi **database**, dan menerima data dari database, misalkan nasabah ingin melihat data transaksinya selama ini. Maka hasil dari server tersebut diterjemahkan sesai dengan aturan bank tersebut,

dilewatkan lagi melalui protocol HTTP, dan akhirnya diterima kembali oleh client.

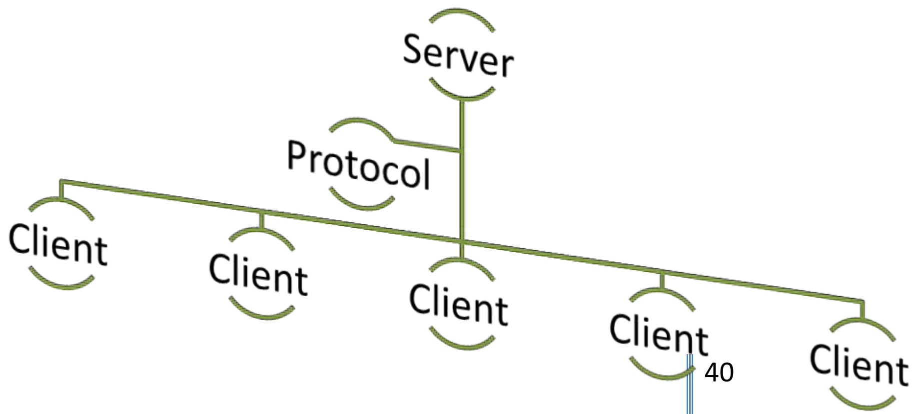
Hubungan client server tidak akan terjadi kecuali client meminta sebuah layanan dari server. Jadi, tidak mungkin server tiba-tiba mengirim client data yang tidak diinginkan oleh client.

Ada berbagai macam aplikasi yang menerapkan model klien, beberapa diantaranya sangat populer.

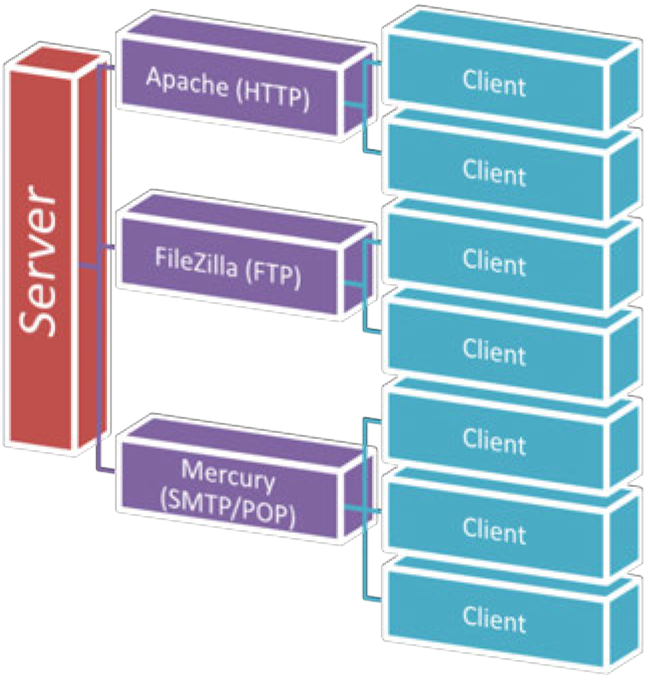
Nama Aplikasi	Protocol
Apache Web Server	HTTP
GlassFish Web Server	HTTP
FileZilla File Server	FTP
Mercury File Server	SMTP/POP
Remote Desktop	RPC
Bind9	DNS

Komunikasi client server hanya bisa terjadi apabila client dan server sama-sama mengikuti protokol yang sama. Perbedaan protocol diibaratkan meskipun sama-sama bisa bicara, tapi itu pembicaraan antara manusia dengan hewan.

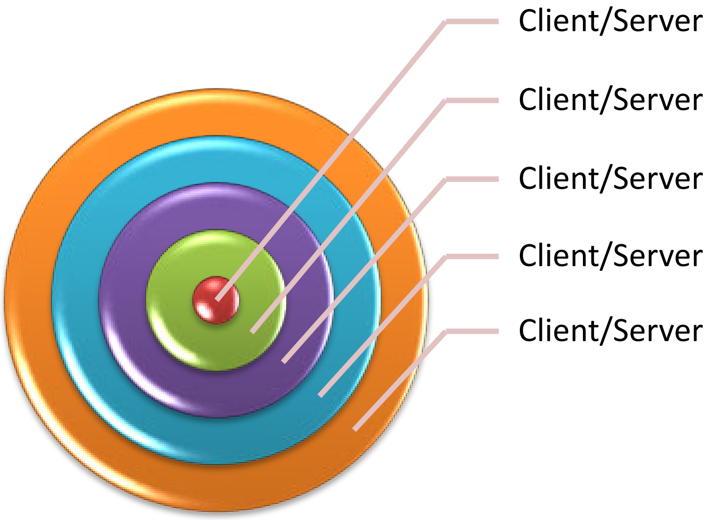
Hirarki Koneksi Client Server



Server dan client bisa berkomunikasi karena adanya protocol diantara mereka. Keberadaan protocol ini menjembatani antara client dengan server, sehingga keseragaman komunikasi bisa tercapai.



Client dan server terhubung dengan bermacam-macam protocol. Meskipun ada banyak protocol, selama server dan client sama-sama memahami protocol tersebut, maka tidak akan ada masalah.



Salah satu model komunikasi lainya adalah peer-to-peer, dimana client menjadi client sekaligus server, berhubungan dengan client lainya sebagai client sekaligus client. Peer-to-peer tetap menerapkan prinsip kerja komunikasi client server. Tapi dalam skala yang lebih kompleks, dan juga lebih besar. Peer-to-peer tidak mempunyai struktur otoritas yang jelas, berbeda dengan client server konvensional yang jelas otoritas tertinggi ada di server utama.

c. Rangkuman

Model komunikasi client server adalah model komunikasi dua arah yang berjalan ketika client meminta sebuah layanan dari server. Model komunikasi client server harus berjalan dengan aturan yang sama yang disebut dengan protokol, protokol memfasilitasi bagaimana client mengirim dan mengambil data dari server, atau server mengirim dan mengambil data dari client.

Semua proses berjalan dua arah, client dan server saling berinteraksi, dan server hanya berjalan ketika client meminta layanan darinya.

Ada berbagai macam protocol, semuanya disesuaikan dengan kebutuhan server dan client, seperti HTTP, FTP, SMTP, POP, dsb. Semua protocol ini

menjembatani antara client dengan server, dan melakukan proses khusus supaya client bisa membaca data dari server dan sebaliknya.

d. Tugas

1. Bagaimana cara kerja model komunikasi client server?
2. Jelaskan proses hubungan komunikasi client server!
3. Sebutkan jenis-jenis aplikasi yang menggunakan model komunikasi client server!
4. Apa keunggulan model client server?
5. Sebutkan dan jelaskan perbedaan model client server dengan model peer to peer!
6. Apakah model client server bisa memiliki lebih dari satu client? Jelaskan!
7. Bagaimana cara client server melayani lebih dari satu client?
8. Dimanakan model client server ini cocok diterapkan?
9. Apa yang dimaksude dengan request-response?
10. Bisakah server mengirim data ke client tanpa client meminta terlebih dahulu? Jelaskan!

e. Test Formatif

1. Berapa jumlah minimal PC yang digunakan untuk menerapkan model client server?
   a. 1 PC
   b. 2 PC
   c. 3 PC
   d. 4 PC
   e. Semua Benar
2. Model client server diterapkan pada layanan-layanan berikut, kecuali ...
   a. HTTP
   b. FTP
   c. DHCP
   d. TORRENT
   e. Semua Salah



3. Urutan komunikasi client server yang benar adalah ...
    a. Client -> Data -> Protocol -> Data -> Server
    b. Client -> Protocol -> Data -> Server
    c. Server -> Data -> Protocol -> Data
    d. Semua Salah
    e. Semua benar
4. Jumlah maksimal client yang bisa diatasi server adalah ...
    a. 1024
    b. 2048
    c. 512
    d. 256
    e. Semua Salah
5. Untuk bisa berhubungan dengan seragam, maka harus ada ...
    a. Client
    b. Server
    c. Protocol
    d. DHCP
    e. HTTP
6. Model komunikasi gabungan di mana semua client juga menjadi server disebut ...
    a. Client Server
    b. HTTP Server
    c. Peer to Peer
    d. Peer on Peer
    e. Server on Server
7. HTTP Server menerapkan model client server karena ...
    a. HTTP Server layanan terpusat di mana semua pengguna terhubung denganya
    b. HTTP Server layanan terpusat di mana satu pengguna terhubung denganya
    c. HTTP Server layanan terpisah di mana dia membutuhkan pengguna



d. HTTP Server layanan terpisah di mana semua pengguna terpisah denganya
e. Semua salah

8. Client harus ... sebelum mendapat layanan dari server
   a. Menggunakan protocol
   b. Memasukkan data
   c. Membuka koneksi
   d. Memanggil data
   e. Semua salah
9. Protocol mengatur ... antara client dan server
   a. Keseragaman komunikasi
   b. Perbedaan komunikasi
   c. Persamaan data
   d. Keseragaman data
   e. Semua salah
10. Server dan client yang berjalan di PC sendiri disebut ...
   a. Wide Server
   b. Local Server
   c. Net Server
   d. HTTP Server
   e. Semua salah

f. Lembar Jawaban Tes Formatif

g. Lembar Kerja Siswa

6. Kegiatan Belajar 6 : Menalar Prinsip Kerja Komunikasi Client Server

a. Tujuan Pembelajaran :Setelah mengikuti kegiatan belajar 6 ini, siswa diharapkan dapat:

a. Menalar prinsip kerja komunikasi client server.
b. **Penalaran Prinsip Kerja Komunikasi Client Server**

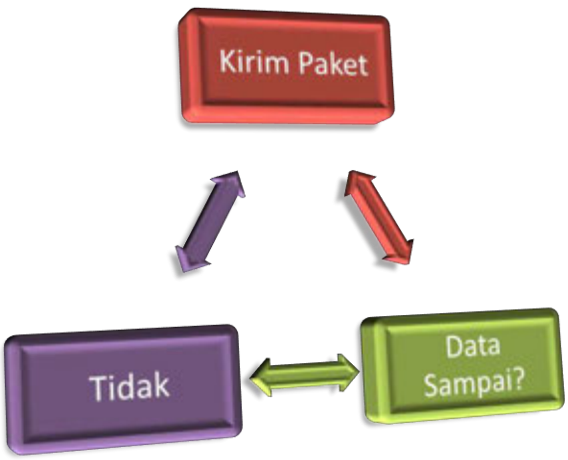
b. Uraian Materi

1. Prinsip Kerja Client Server

Prinsip kerja client server, pada dasarnya juga bergantung dengan lapisan di bawahnya, yaitu transport. Komunikasi antara client bergantung pada lapisan ini.

Komunikasi yang lebih rendah ini menggunakan packet sebagai bentuk data yang dikirim, paket ini diberikan header yang mengindikasikan informasi tujuan dan asal.

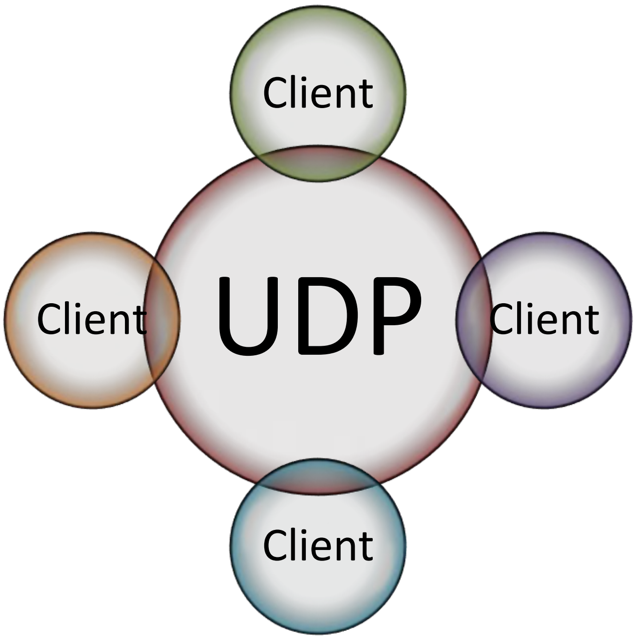
Protocol untuk pengiriman paket bermacam-macam, yang populer adalah TCP, dan UDP. Biasanya, komunikasi client server bersifat penting, dan tidak boleh ada data yang hilang, maka protocol TCP yang digunakan. Kenapa? Karena TCP melakukan proses tanya jawab, TCP memastikan target menerima pesan dari asal.



TCP mengirimkan paket, lalu menunggu tanda dari target, apakah dia menerima paket tersebut. Apabila tidak, maka TCP akan mengulangi mengirim paket tersebut. Begitu seterusnya, sampai paket yang dikirim sampai tujuan semuanya. Namun, karena proses tanya-jawab antara asal dan target mengenai sampainya paket ini terus terjadi, TCP lebih lambat dan memakan resources lebih besar.



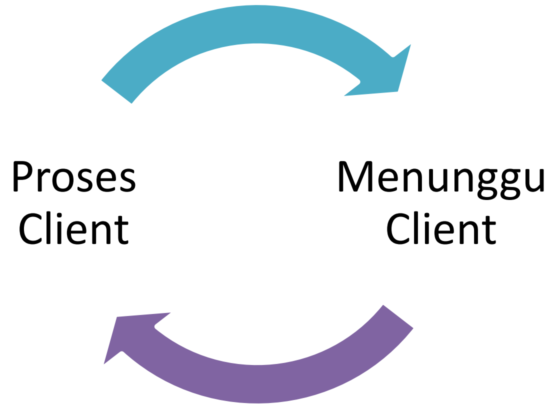
Berbeda dengan UDP. UDP tidak peduli apakah data sudah terkirim dan diterima oleh target. UDP sangat cocok untuk komunikasi yang tidak begitu penting, misalkan dalam sebuah game, dimana client dan server saling memberitahukan keadaan pemain sekarang. Karena UDP tidak begitu peduli, maka kejadian seperti LAG bisa terjadi. Namun, UDP tidak melakukan proses tanya jawab seperti TCP, sehingga komunikasi terjadi lebih cepat.



UDP hanya menyebarkan informasi, tidak peduli apakah client menerimanya dengan sempurna atau tidak.

Lalu bagaimana komunikasi client – server berlangsung? Kita akan ambil contoh, proses komunikasi client server antara web browser dengan web server.

Web server, sebagai penyedia halaman web, dinyalakan. Dia hanya diam dan menunggu untuk kedatangan client.



Web server akan terus dalam posisi menunggu sampai ada client yang meminta layanan darinya.

Web browser dan web server sama-sama mempunyai protocol yang sama, yaitu HTTP. HTTP kependekan dari *Hyper Text Transfer Protocol* adalah protocol untuk bertukar informasi dalam bentuk hyper text.

Bagaimana protocol HTTP itu? Protocol HTTP mempunyai 2 bagian, header dan content. Bagian header untuk meminta data dari server berbeda dengan header untuk mengirim dari server.

```
POST /index.php HTTP/1.1
Content-Type: application/x-www-form-urlencoded
Content-Length:4

Hello
```

Bagaimaa HTTP memisahkan antara bagian header dan content? Jawabanya dengan baris kosong. Antara content dan header, ada sebuah baris kosong.



Header HTTP digunakan untuk memberikan informasi tentang content. Jadi ketika server membacanya, dia tahu bahwa client ingin mengirim data dengan metode POST, ke halaman index.php dengan protocol HTTP versi 1.1.
Server mengetahui bahwa data yang dikirim sepanjang 4 bytes, dan akhirnya server mendeteksi adanya baris kosong, inilah saatnya server membaca data yang masuk bukan sebagai header lagi, tapi sebagai content.

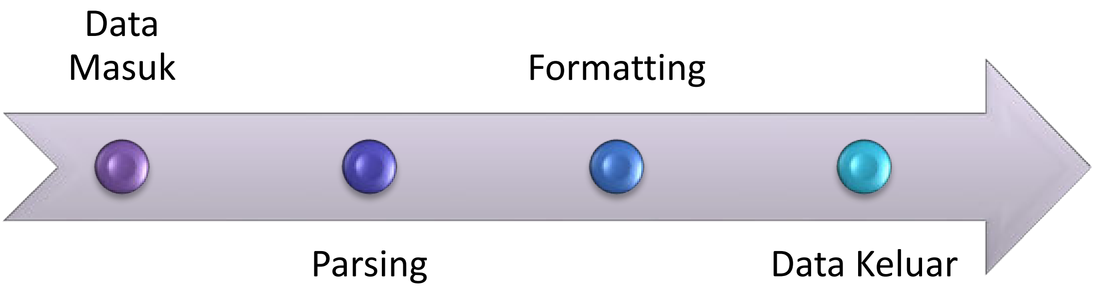
Server akan melakukan proses data, menjalankan program untuk mengakses database apabila diperlukan, melakukan akses ke berbagai berkas di server apabila diperlukan, hingga akhirnya server mendapatkan sumber daya atau hasil yg bisa diberikan kepada client.

Akhirnya, masih dengan menggunakan protocol HTTP, server mengirim kembali data hasil pemrosesan tadi.

```
HTTP/1.1 200 OK
Content-Type: text/html
Content-length: 7

Success
```

Web browser akan menganalisa hasil keluaran dari server, dia aka membaca bahwa server memberikan balasan versi protocol HTTP 1.1. Kode HTTP 200, berarti OK, server memproses permintaan dengan lancar, tidak ada kesalahan. Sisanya server memberi tahu informasi tentang content utama yang diminta.



Cara penerapan protocol, data masuk akan di*parsing* atau diterjemahkan. Program membaca data yang masuk dan mengambil informasi yang dibutuhkan.

Setelah data yang penting didapatkan, maka program melakukan penyesuaian format, sehingga hasil dari protocol tersebut bisa seragam. Keseragaman hasil protocol inilah yang membuat client server memahami protocol masing-masing.

```
GET /index.php HTTP/1.1
```

Keseragaman ini bisa dilihat di protocol HTTP, dimana semua data yang diberikan akan dirubah sesuai format protocol HTTP, yaitu adanya header, dan adanya content.

Server atau client melakukam hal yang sama, mereka membuat header dan juga content sesuai dengan data yang sedang mereka proses.

c. Rangkuman

Mencoba menalar bagaimana prinsip kerja client server, kita mengetahui bagaimana kerja client server dengan menelusuri lapisan yang mendukungnya. Yaitu lapisan transport. Client server biasanya menggunakan protocol TCP untuk transportnya, meskipun protocol lainya juga tidak dipungkiri bisa digunakan.

TCP menyebarkan informasi ke client dengan handal, tidak boleh ada data yang tertinggal. Berbeda dengan UDP yang tidak peduli apakah ada data yang tertinggal atau tidak.

Komunikasi client server harus berjalan di atas protocol yang sama, protocol ini mengambil data, melakukan penerjemahan, melakukan formating, dan mengembalikan keluaran data yang sudah seragam sehingga antara client dan server sama-sama bisa mengerti isi data tersebut.

Salah satu protocol tersebut adalah HTTP, yang setelah data diterjemahkan dan diformat, hasilnya adalah sebuah struktur data dengan header dan content. Header menjelaskan isi dari content, dan content berisi data yang dikirim atau diterima.

HTTP menghasilkan keluaran yang seragam, oleh karena itu antara web browser dan web server bisa bekerja sama untuk mendapatkan hasil yang diinginkan.

d. Tugas

1. Siswa ditugaskan untuk membuat kelompok yang terdiri dari 3 – 5 orang.
2. Siswa ditugaskan untuk mempelajari komunikasi yang terjadi pada HTTP server.
3. Siswa ditugaskan untuk membuat protokol sederhana yang bisa disimulasikan dengan gerakan tubuh dengan model client server.
4. Siswa membuat laporan analisa dari cara client server bekerja.
5. Siswa menyajikan laporan dalam bentuk presentasi yang singkat, padat, dan jelas.

e. Tes Formatif

f. Lembar Jawaban Tes Formatif

g. Lembar Kerja Siswa

7. Kegiatan Belajar 7 : Memahami Instalasi Sistem Operasi Server

a. Tujuan Pembelajaran: Setelah mengikuti kegiatan belajar 7 ini, siswa diharapkan dapat:

c. Memahami cara instalasi sistem operasi server linux.

b. Uraian Materi

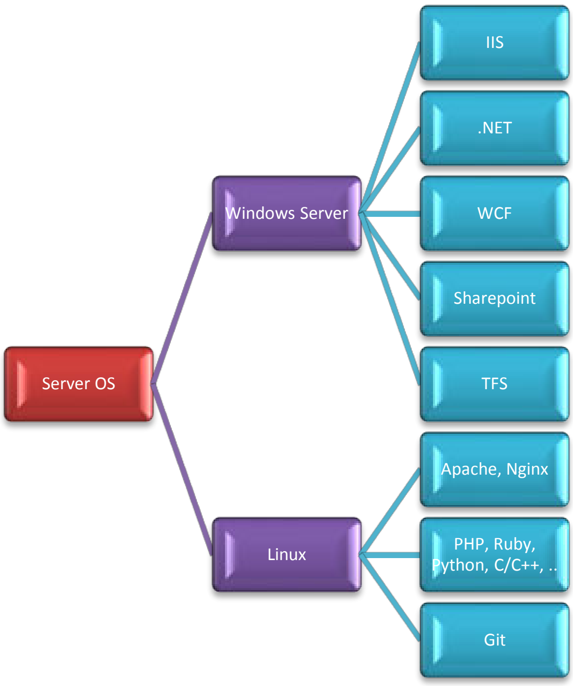
d. **Memahami Instalasi Sistem Operasi Server**

2. Penjelasan Sistem Operasi Server

Sistem operasi server, tidak berbeda layaknya seperti sistem operasi desktop biasa. Mereka bisa dipasang berbagai macam aplikasi, digunakan untuk manipulasi teks, bermain game. Hanya saja, mereka dikhususkan untuk menangani jaringan lebih cepat dari biasanya, dengan mengorbankan beberapa fitur sistem operasi desktop.

Kenapa harus menggunakan sistem operasi server? Pasalnya, sistem operasi server telah dikhususkan untuk keperluan jaringan, kemampuan mereka sudah dioptimalkan untuk mengatasi hubungan dengan jaringan. Seperti multi-user, kemanaan, stabilitas dan kolaborasi.

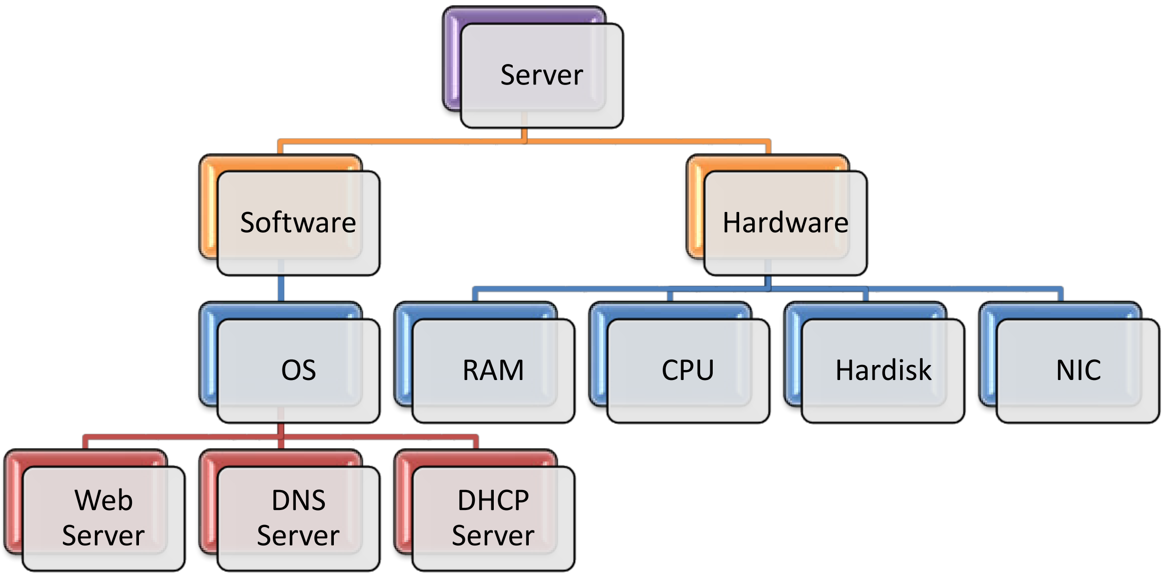
Ada berbagai macam sistem operasi server di luar sana, yang menguasai pasar populer menengah adalah Windows Server dan Linux.



Pembeda antara server Linux dan Windows Server adalah Windows Server dilengkapi dengan software-software komersil dari Microsoft. IIS sebagai web server, .NET sebagai platform (yang berarti mendukung semua bahasa pemrograman yang berjalan di atas .NET), WCF sebagai web service, sharepoint untuk kolaborasi, dan Team Foundation Server untuk repository dan kerja tim.

Bukan berarti Windows Server hanya terbatas dengan aplikasi di atas, baik Linux atau Windows Server bisa ditambahkan software lainya yang mendukung kerja mereka.

Linux, hampir semua software bisa dipasang di sini, bedanya software yang dikhusukan untuk Windows Server tidak akan bisa dipasang di sini, terlebih lagi

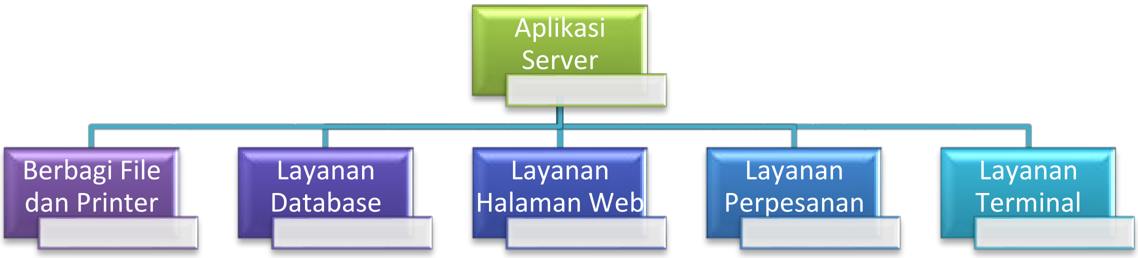


software komersil dari Microsoft.

Arsitektur server sederhana, server terdiri dari 2 komponen utama, software dan hardware. Software terdiri dari sistem operasi, sistem operasi bisa menampung aplikasi-aplikasi web server, DNS server, DHCP server, dll. Hardware menampung perangkat-perangkat keras di mesin server.

Apa saja tugas atau aplikasi yang bisa di pasang di sistem operasi server?

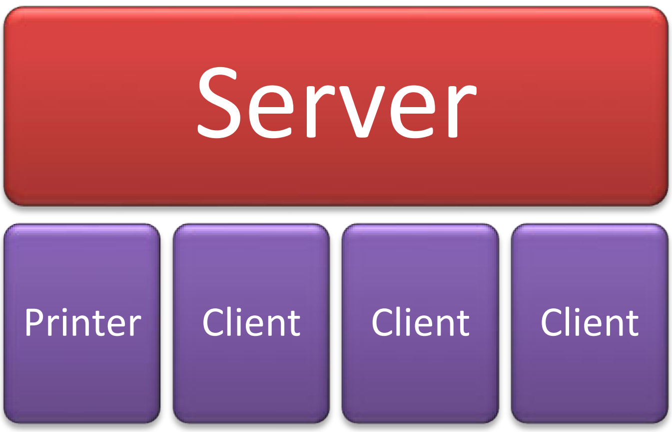




Fitur-fitur yang biasa didukung oleh sistem operasi server, yang bisa dipasang didalamnya adalah seperti aplikasi untuk berbagi file dan printer, layanan database, web, perpesanan, atau terminal.

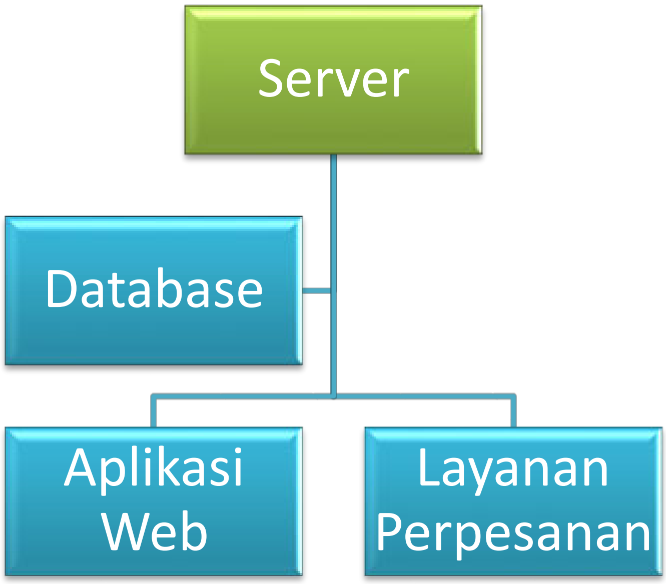
Berbagi File dan Printer

Layanan ini memungkinkan banyak pengguna bisa berbagi file yang berada di server, mereka juga bisa menggunakan satu printer bersama-sama untuk melakukan suatu pekerjaan tanpa harus melakukan koneksi langsung dengan printer.



Layanan Database

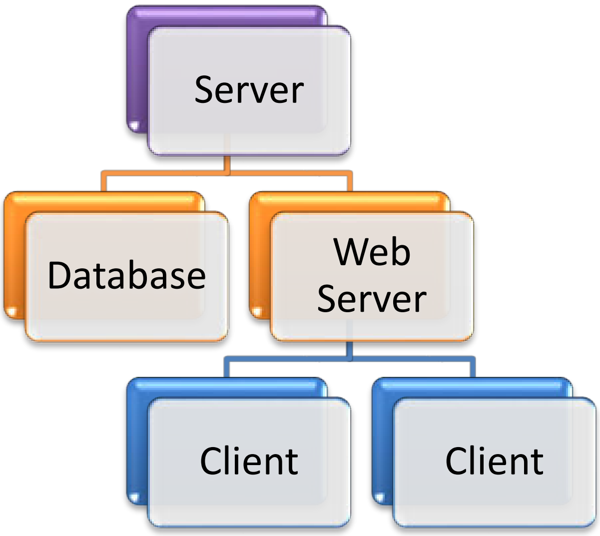
Database berfungsi untuk menyimpan data, penggunaan layanan database biasanya difasilitasi software pihak ketiga, yang menggunakanya untuk mempermudah akses ke dalam database.



Layanan Halaman Web

Halaman web, dengan protocol HTTP disediakan bagi pengguna untuk melihat sebuah halaman HTML. Layanan ini biasanya juga menggunakan layanan database, dengan menggunakan bahasa pemrograman yang bisa berjalan di atas web server.

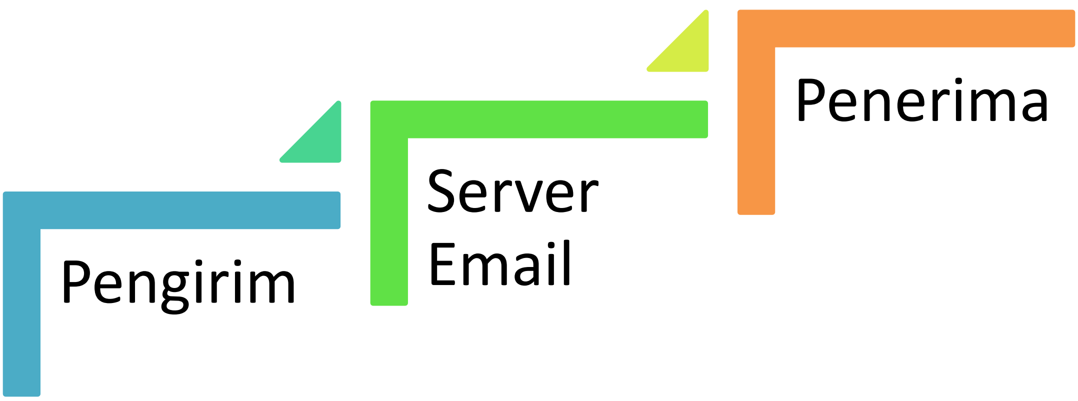




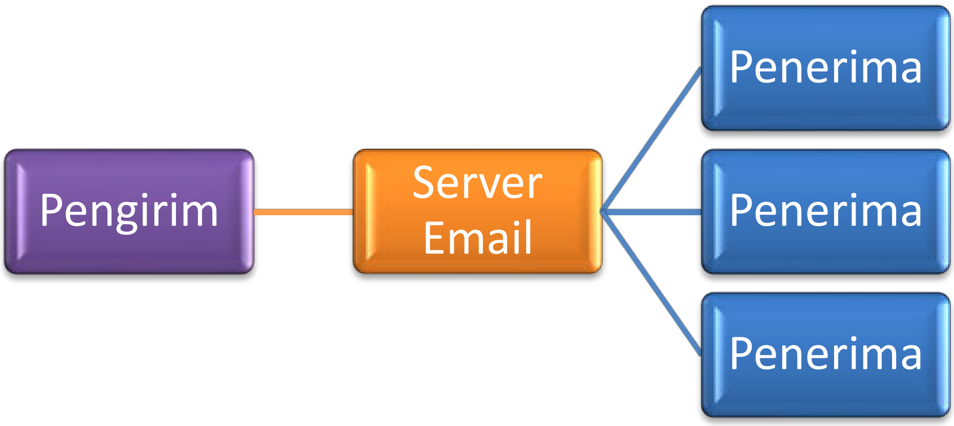
Layanan Perpesanan

Mengirim e-mail, group e-mail, atau melakukan perpesanan biasa bisa difasilitasi oleh sistem operasi server. Pengguna bisa mengirim pesan terhadap satu orang, atau menggunakan group e-mail dan mengirim pesanya ke semua orang dalam group tersebut, atau chatting dengan pengguna lainya secara langsung.

E-mail



E-mail Group

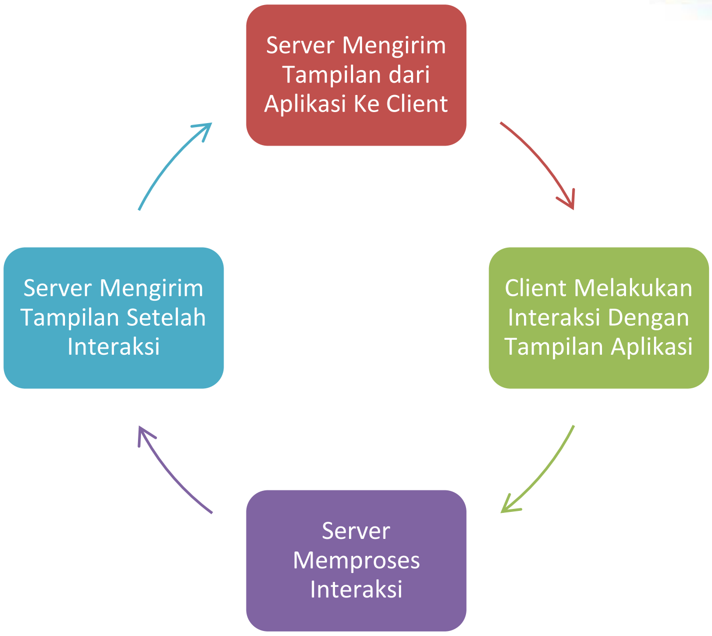


Layanan Terminal

Layanan terminal, berfungsi untuk menjalankan aplikasi di server. Client hanya menerima tampilanya saja, semua gerakan client, seperti mouse, dan keyboard terkirim ke server dan server memprosesnya.

Seorang client membuka Microsoft Word, maka server hanya memberikan tampilan Microsoft Word, ketika client mengetikkan huruf 'A', maka data terkirim ke server dan server yang melakukan proses sebenarnya. Setelah proses selesai, server memberikan tampilan Microsoft Word setelah proses kembali kepada client.





3. **Memilih Sistem Operasi**

Memilih sistem operasi yang tepat menentukan kinerja server kita. Salah satu server yang mudah dipelajari dan mudah didapat, didukung komunitas yang besar dan gratis, bisa digunakan untuk belajar adalah Linux.

Sebelum memilih server, kita harus memberikan kriteria apa saja yang kita butuhkan. Biasanya, yang dicari saat menentukan sistem operasi adalah,

Administrasi	•Seberapa mudah mendapatkan perangkat •Sebarapa mudah cara menggunakan perangkat
Keamanan	•Menutup celah untuk peretas •Memberikan kepribadian kepada pengguna
Stabilitas	•Seberapa kuat server berjalan tanpa berhenti
Fitur	•Apakah ada aplikasi yang dibutuhkan (Misal Web Server, FTP Server)
Scalabilitas	•Apakah sistem operasi bisa mengadaptasi perubahan di masa mendatang? Hardware baru, software baru?
Aplikasi Pihak Ketiga	•Apakah ada aplikasi bukan bawaan server yang tersedia untuk memenuhi kebutuhan?

Apabila kita hendak membuat aplikasi perusahaan besar, maka sistem operasi gratis sudah tidak bisa lagi mendukung kebutuhan kita. Dibutuhkan sistem operasi yang memang ditujukan untuk perusahaan besar, seperti Solaris, IRIX, AIX, HP-UX, dll.

Untuk media pembelajaran, sistem operasi gratis seperti Linux sudah lebih dari cukup untuk memehuni kebutuhan kita.

4. **Lisensi Server**

Ada berbagai macam jenis lisensi yang diberikan saat memilih server. Server gratis seperti Linux, tidak perlu kerumitan untuk menentukan lisensinya. Tapi, server berbayar seperti Windows Server, atau Solaris membutuhkan lisensi yang tentunya tidak semudah lisensi gratis.

Beberapa dari vendor sistem operasi menerapkan jumlah pengguna yang menggunakan server sebagai penentuan lisensinya, seperti Windows Server.

Vendor lainya menentukan lisensi berdasarkan jumlah CPU yang terpasang di server. Windows Server juga menggunakan lisensi alternatif ini, begitu juga dengan Solaris.

Memilih lisensi server yang tepat menentukan faktor skalabilitas dari server yang dibuat. Lisensi dengan menghitung jumlah CPU, menuliskan di perjanjian lisensinya bahwa jumlah pengguna yang menggunakan server tidak terbatas. Jenis lisensi ini sangat fleksibel karena biasanya jumlah pengguna berubah lebih cepat daripada jumlah CPU atau spesifikasi hardware.

5. **Pemasangan Sitem Operasi Server Linux Debian**

Untuk pembelajaran, sistem operasi server Linux sudah cukup. Ada beberapa varian dari sistem operasi Linux, seperti Ubuntu, CentOS, Fedora, Debian, dll. Debian merupakan salah satu sistem operasi yang

sudah cukup lengkap, dilengkapi dengan berbagai aplikasi server, dan bisa digunakan dengan mudah dan gratis.

Instalasi sistem operasi server hampir sama dengan instalasi sistem operasi biasa.

- ✓ Menentukan lokasi instalasi.
- ✓ Alokasi partisi.
- ✓ Menentukan informasi akun utama (root).
- ✓ Menentukan informasi tentang server.

Persiapan Instalasi

Sebelum melakukan instalasi, ada beberapa perlengkapan yang harus dimiliki. Satu PC yang akan dijadikan server, juga master Debian. Master Debian bisa didapatkan dengan cara membeli di toko komputer atau jasa persewaan software.

Bagi yang ingin mengunduh sendiri master Debian bisa dari

```
http://cdimage.debian.org/debian-cd/7.2.0/i386/iso-dvd/
```

Setelah mengunduh, pastikan mempunyai minimal 3 DVD kosong, lalu bakar hasil unduhan Debian ke tiap DVD.

Setelah semua perlengkapan siap, kita bisa melanjutkan ke proses instalasi Debian.

c. Rangkuman

Sistem operasi server merupakan sistem operasi yang digunakan untuk server, setiap kegiatan yang server lakukan ditujukan untuk operasi pada jaringan. Sistem operasi server dioptimalkan untuk bekerja secara stabil, dan cepat dalam mengatasi pekerjaan kritikal.



Sistem operasi server bekerja secara kritikal, harus terus menyediakan layanan pada pengguna. Pemilihan sistem operasi yang tepat menentukan kekuatan server untuk melayani pengguna.

Sistem operasi server merupakan pondasi awal dari sistem server. Di atas sistem operasi server bisa dipasang aplikasi yang mendukung kebutuhan sistem, seperti web server, ftp, dns, dsb.

Sistem operasi dipilih berdasarkan kemampuan administrasi, keamanan, stabilitas, fitur, skalabilitas, dan dukungan aplikasi pihak ketiga. Dengan aplikasi ini, sistem operasi bisa memberikan kemampuan seperti berbagi file dan printer, melayani penyimpanan data, layanan web, perpesanan, terminal, dsb.

Vendor server menetapkan lisensi masing-masing, pemilihan lisensi yang tepat mempengaruhi faktor skalabilitas server kita. Lisensi tiap vendor berbeda-beda, ada yang menetapkan berdasarkan jumlah pengguna, ada yang menetapkan berdasarkan jumlah CPU, dsb.

d. Tugas

1. Apa itu sistem operasi server?
2. Kenapa harus menggunakan sistem operasi server?
3. Apa saja tugas sistem operasi server?
4. Apa saja yang bisa dipasang di sistem operasi server?
5. Apa saja yang harus diperhatikan dalam memilih sistem operasi? Sebutkan dan jelaskan!
6. Apa yang harus diperhatikan dalam menangani lisensi server?
7. Sebutkan dan jelaskan langkah-langkah installasi sistem operasi Debian!
8. Apa nama user utama di sistem operasi Debian?
9. Apa perbedaan mencolok dari sistem operasi Linux dan Windows? Jelaskan!
10. Sebutkan kelebihan dan kekurangan dari sistem operasi Debian!



e. Test Formatif

i. Apa saja yang bisa dipasang di sistem operasi server?
   1. Permainan
   2. Aplikasi
   3. Layanan
   4. Pengolah Data
   5. Semua benar

ii. Apa kelebihan sistem operasi Windows dibanding Linux?
   1. Bisa dipasang Apache
   2. Bisa dipasang PHP
   3. Bisa dipasang WCF
   4. Bisa dipasang Ruby
   5. Salah semua

iii. Letak layanan DHCP, DNS, Web, dsb terletak di ...
   1. Sistem Operasi
   2. Hardware
   3. RAM
   4. Memory
   5. Semua benar

iv. Berikut merupakan layanan yang biasa diberikan oleh sistem operasi, kecuali ...
   1. Layanan berbagi file dan printer
   2. Layanan database
   3. Layanan terminal
   4. Layanan energi
   5. Layanan perpesanan

v. Apa yang terjadi saat dalam proses remote terminal, lalu pengguna menekan tombol proses?
   1. Aplikasi pengguna berjalan dan mengirimkan hasil proses ke server
   2. Aplikasi pengguna mengirimkan indikasi bahwa tombol proses di tekan ke server
   3. Aplikasi pengguna mengirim tombol proses ke server
   4. Aplikasi pengguna memproses data dari server
   5. Semua salah



vi. Salah satu faktor yang dipertimbangkan untuk pemilihan sistem operasi, kecuali ...
   1. Administrasi
   2. Stabilitas
   3. Scalabilitas
   4. Probabilitas
   5. Keamanan

vii. Yang bukan termasuk langkah installasi sistem operasi adalah ...
   1. Menentukan lokasi installasi
   2. Alokasi partisi
   3. Menentukan informasi akun utama
   4. Menentukan informasi tentang server
   5. Menentukan informasi tentang lisensi

viii. Pengguna utama dari sistem operasi debian adalah
   1. boot
   2. root
   3. loot
   4. admin
   5. administrator

ix. Kelemahan dari sistem operasi Linux Debian adalah ...
   1. Tidak bisa dipasang software .NET
   2. Tidak bisa dipasang software Apache
   3. Tidak bisa dipasang software Git
   4. Tidak bisa dipasaing software C++
   5. Semua salah

x. Fungsi dari GRUB adalah ...
   1. Membuat sistem operasi menyala
   2. Sebagai pengatur boot record dari sistem operasi
   3. Sebagai sistem operasi cadangan
   4. Sebagai pembantu sistem operasi saat installasi semua salah

f. Lembar Jawaban Tes Formatif

g. Lembar Kerja Siswa

8. Kegiatan Belajar 8 : Menyajikan Hasil Instalasi Sistem Operasi Server

a. Tujuan Pembelajaran

Setelah mengikuti kegiatan belajar 8 ini, siswa diharapkan dapat:

e. Melakukan instalasi sistem operasi server Linux Debian.

f. Mengoperasikan sistem operasi server Linux Debian tingkat dasar.

b. Uraian Materi

g. **Melakukan Instalasi Sistem Operasi Server Linux Debian**

6. Instalasi Debian

Pastikan DVD Debian sudah siap, dengan PC yang akan dijadikan server.

a. Masukkan DVD Debian ke DVD-ROM PC.

b. Masuk BIOS, pastikan PC boot ke DVD-ROM terlebih dahulu.

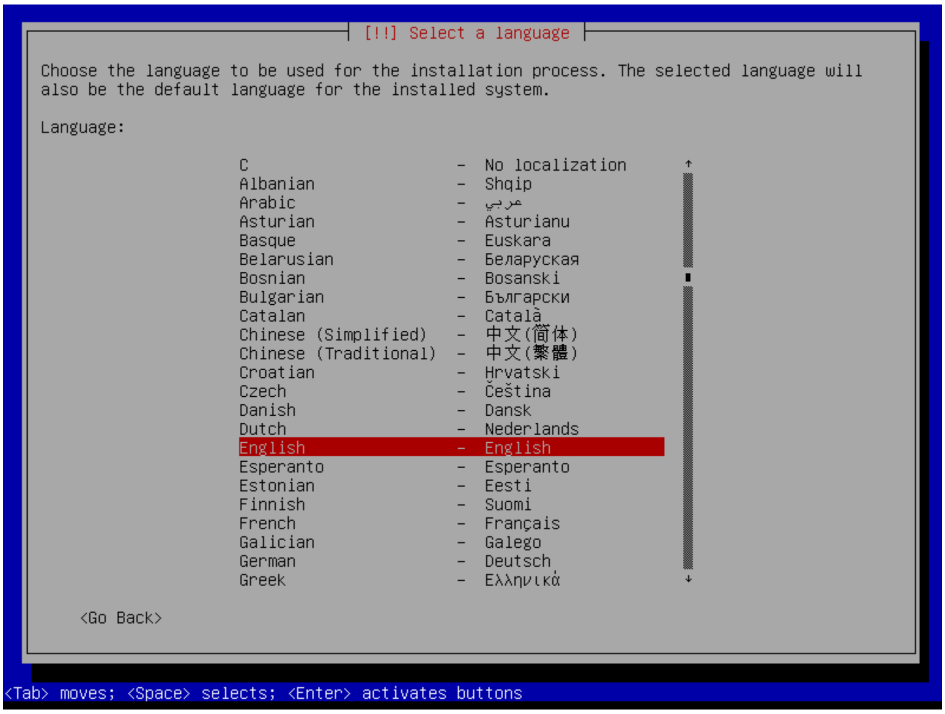


c. Akan muncul tampilan seperti berikut,

Karena kita menginstall untuk server, dan supaya proses instalasi lebih cepat, kita tidak menggunakan instalasi grafik, tapi dengan instalasi *command line*.

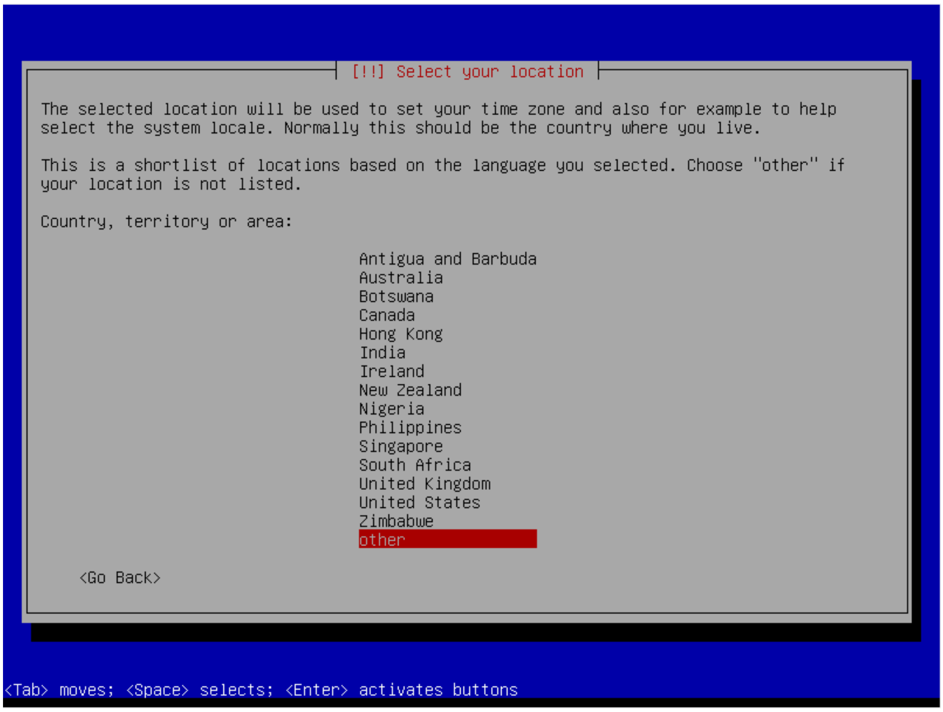


d. Pilih bahasa yang akan digunakan saat proses instalasi, bahasa yang dipilih juga akan menjadi bahasa yang digunakan sistem.



Pilih english, sebagai bahasa instalasi.

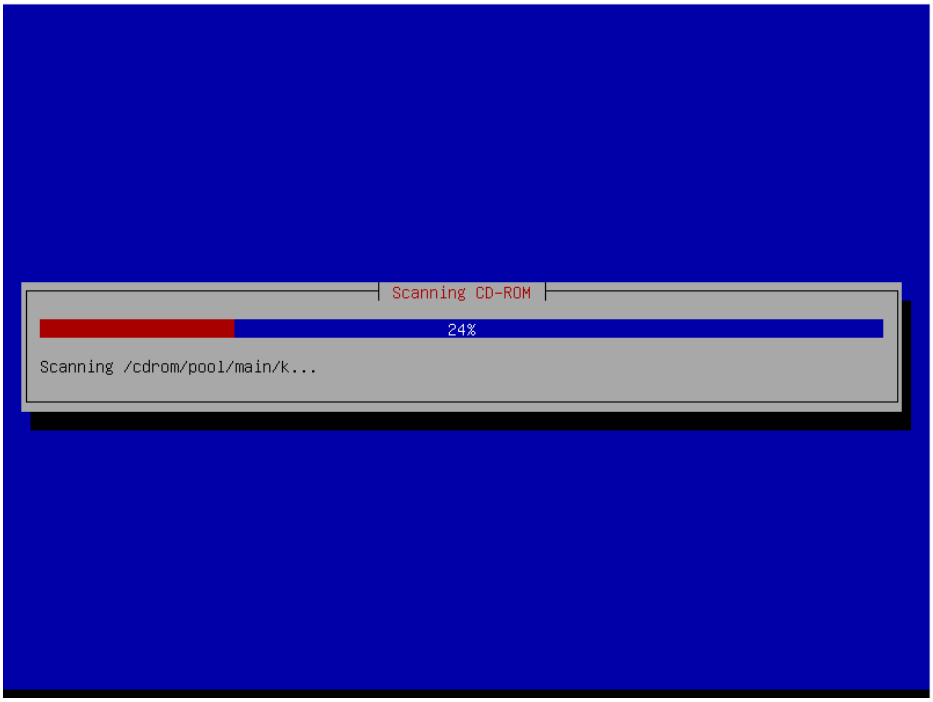
e. Tentukan lokasi negara, Indonesia berada di bagian other | Asia | Indonesia.



Pilih Indonesia sebagai negara, akan muncul pemilihan *locale* atau jenis huruf yang didukung. Indonesia menggunakan alphabet latin, gunakan locale Amerika Serikat.

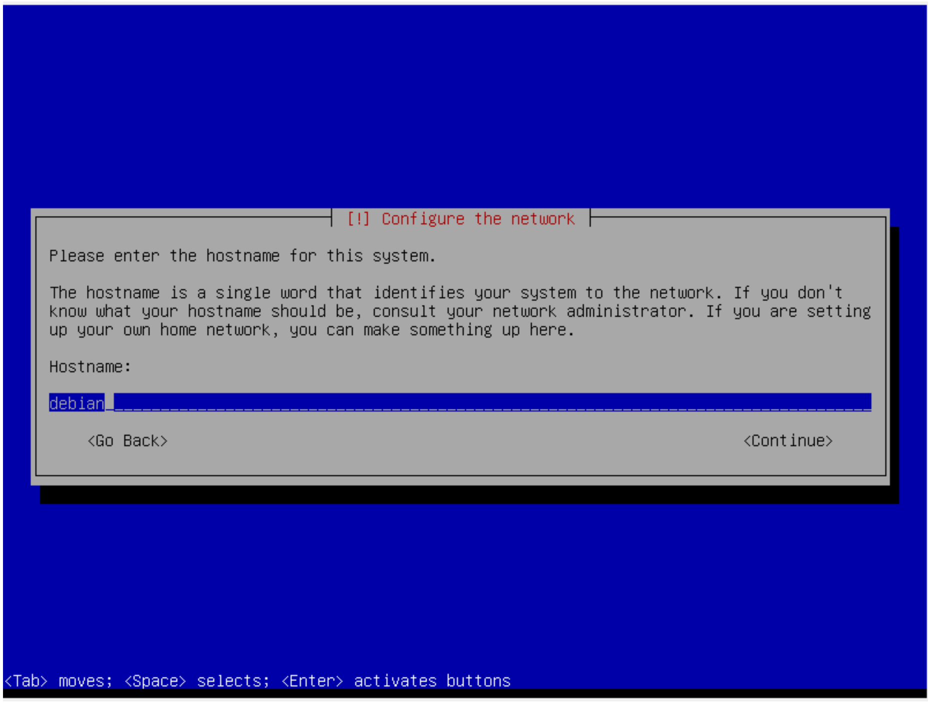
Pilih juga *keymap*, keymap adalah tatanan keyboard yang digunakan. Gunakan **American English**.

f. Tunggu, akan ada proses.



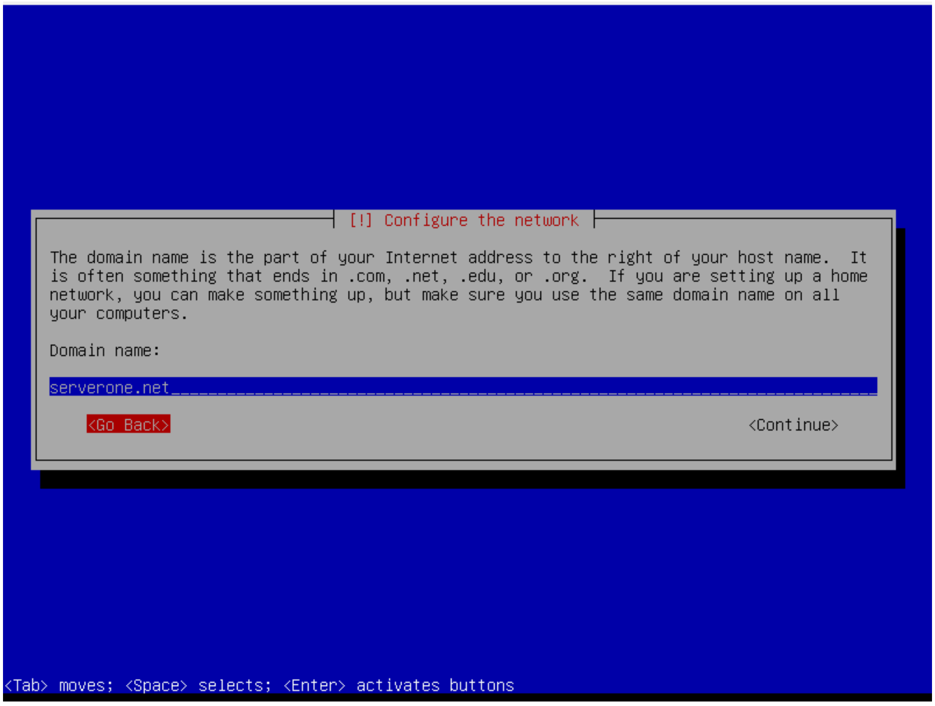
Biarkan prosesnya sampai selesai.

g. Masukkan hostname yang dibutuhkan, karena kita mensetting server kita sendiri. Masukkan sesuai keinginan.



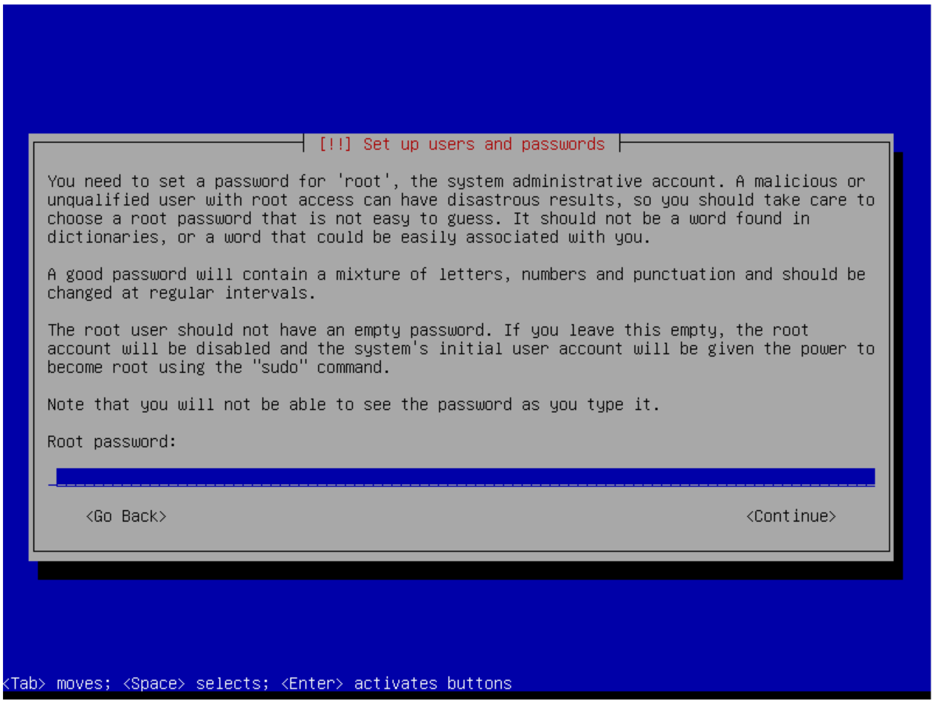
Dalam kasus ini, masukkan **serverone** sebagai hostname.

h. Setelah itu kita akan diminta memasukkan nama domain, karena kita mensetting server kita sendiri, masukan sesuai keinginan.



Dalam kasus ini, masukkan **serverone.net**.

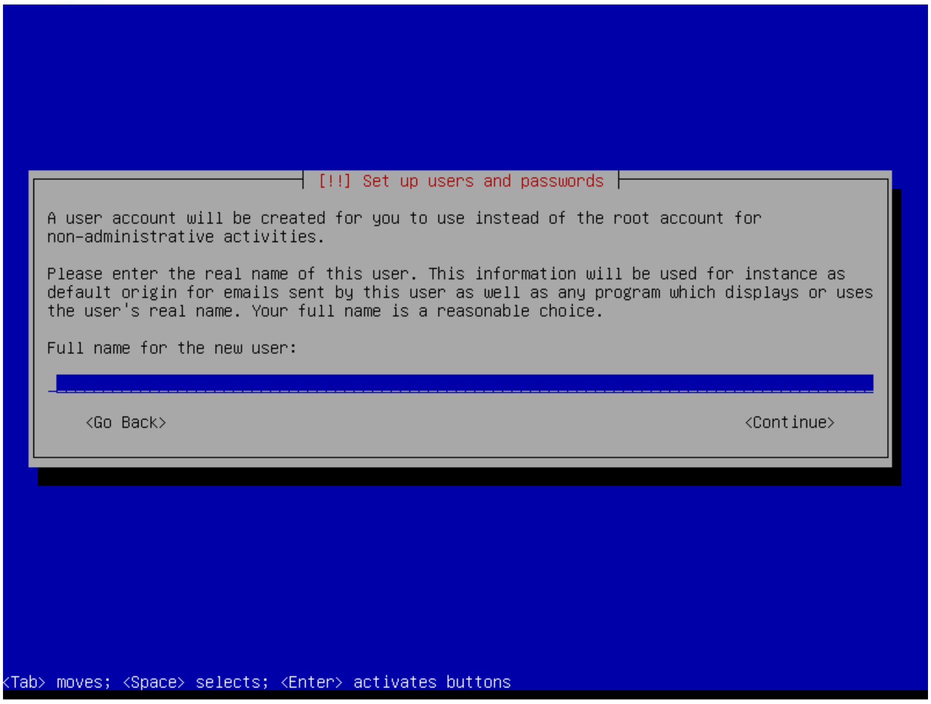
i. Kita akan memasukkan kata sandi untuk akun utama, masukkan sesuai kenginan. Kata sandi yang baik adalah kata yang tidak ada di dalam kamus, terdiri dari huruf dan angka, atau simbol-simbol tertentu, dan dirubah secara berkala.



Dalam kasus ini, masukkan **root**, lalu masukan **root** lagi saat konfirmasi sandi.

Akun utama atau root, memiliki hak akses paling tinggi. Dia bisa melakukan apa saja dengan sistem dan tidak ada yang membatasinya. Pastikan ketika anda membuat server asli, kata sandi akun root anda sangat kuat.

j. Setelah itu, akan diminta untuk memasukkan nama lengkap pengguna. Masukan nama anda.



k. Masukkan nama pengguna untuk akun pribadi anda sendiri. Nama depan merupakan plihan yang cukup. Masukkan juga kata sandi, lalu konfirmasi lagi kata sandi.

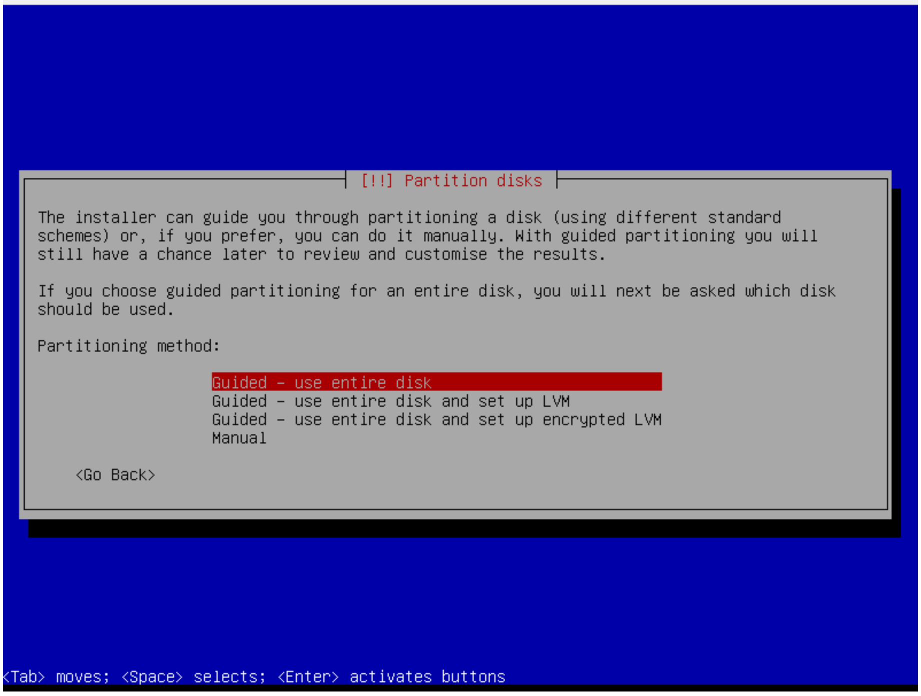
l. Masukkan kota tempat anda tinggal, digunakan untuk menentukan waktu.



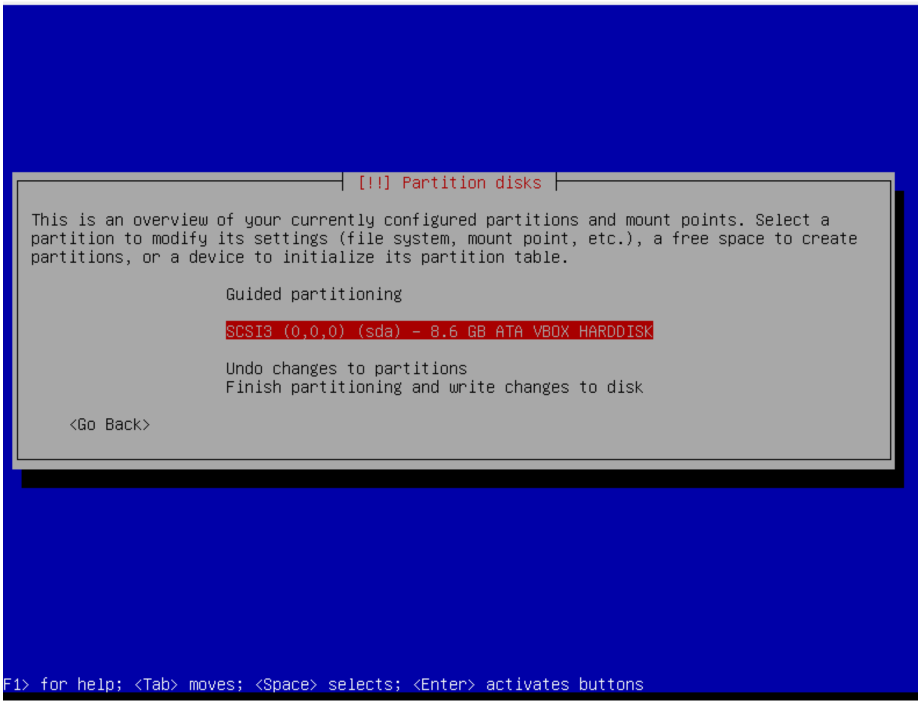
m. Tunggu sampai proses pengecekkan hardware selesai.

n. Pilih proses partisi secara manual,



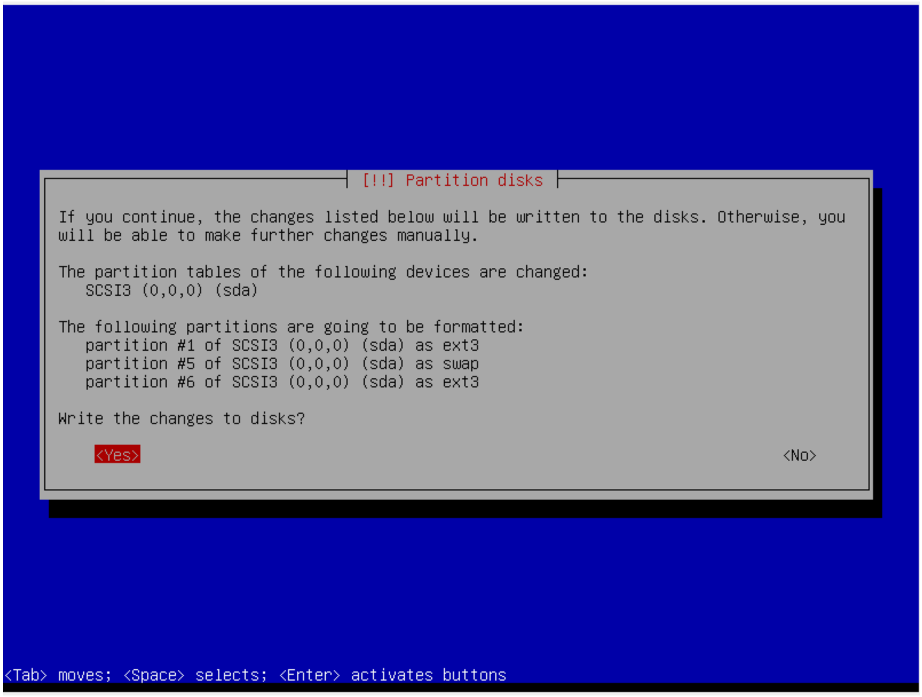


Pilih disk yang akan di partisi, masukkan 50% sebagai ukuran disk.



Tekan enter, dan pilih yes, lihat bagian bootable flag, aktifkan bagian bootable flag.

Lakukan hal yang sama pada sisa partisi lainya.

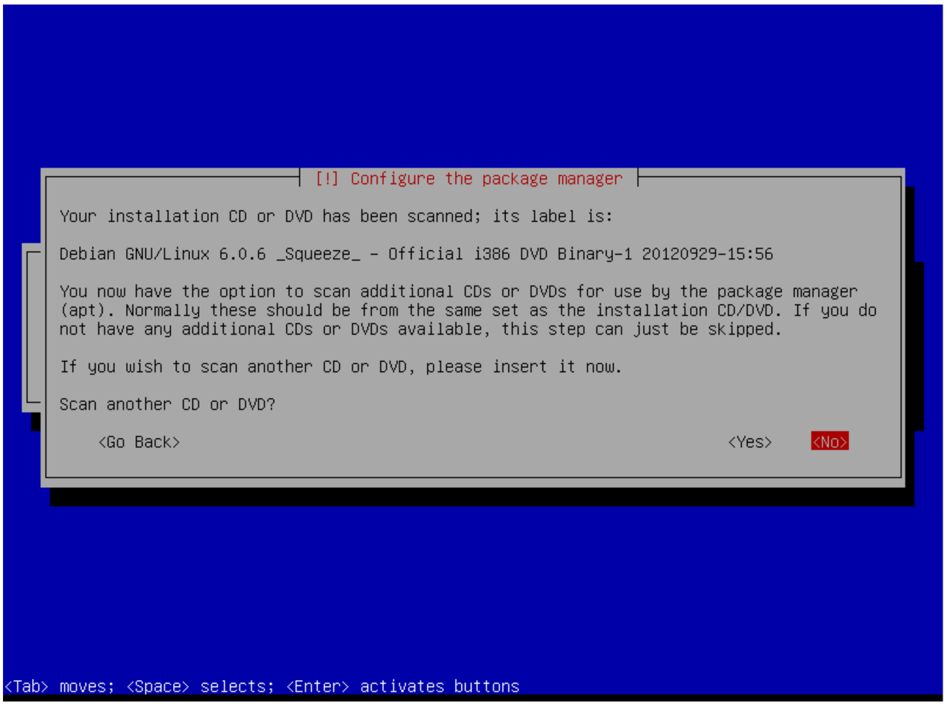


Dalam kasus ini, di server contoh menggunakan 3 partisi. Satu root, satu swap, dan satu home. Pilih yes, lalu biarkan proses partisi berjalan.

o. Biarkan proses instalasi berjalan.



p. Apabila dalam proses instalasi ada pertanyaan,



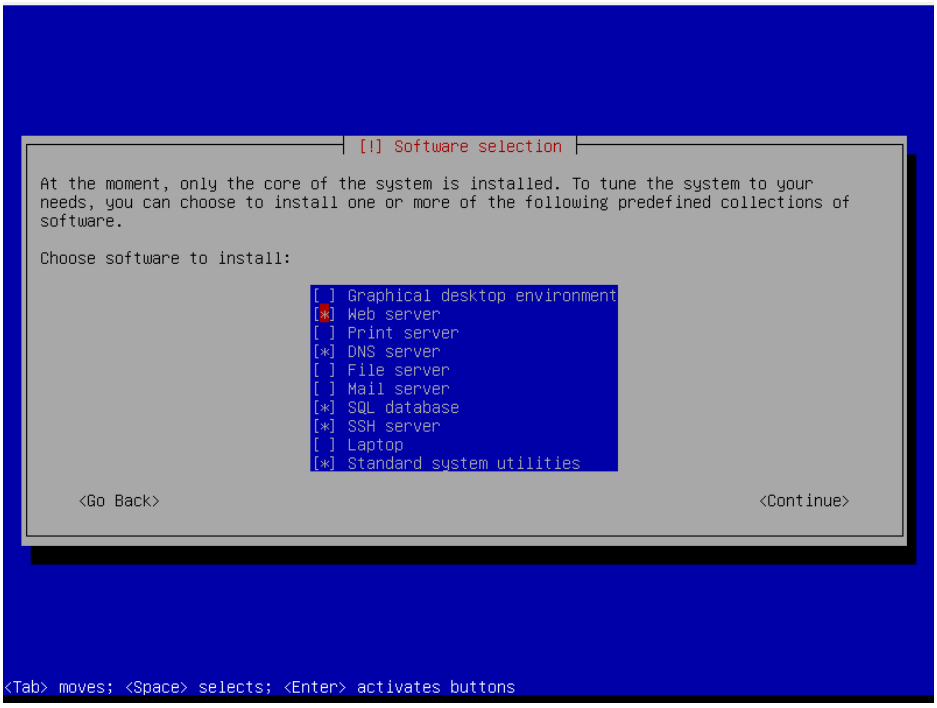
Apakah ada media instalasi lainya? DVD atau CD? Apabila ada, masukkan DVD dan pilih yes, apabila tidak, pilih no.

q. Akan ada pertanyaan lagi, apakah perlu menggunakan network mirror? Network mirror adalah master debian yang ada di internet, berfungsi untuk mengambil data-data yang mungkin tidak ada di DVD atau CD.

Silahkan pilih ya apabila anda ingin menggunakan network mirror, syaratnya anda harus mempunyai koneksi internet.

r. Apabila ada pertanyaan seputar berkontribusi untuk popularity contest, pilih no.

s. Setelah itu akan disediakan pilihan apa saja yang akan diinstal.



Karena kita membuat sebuah server sederhana, silahkan pilih web server, DNS server, SQL Database, SSH Server, dan Standard System Utilities.

Lalu biarkan proses instalasi berjalan kembali.

t. Akan ada pesan, apabila PC yang sudah anda sudah ada sistem operasinya. Maka menginsall GRUB akan embuat sistem operasi yang sebelumnya tidak bisa digunakan. Kecuali nanti diatur ulang.

Kita asumsikan bahwa anda belum menginstall apa-apa karena tujuanya adalah membuat PC khusus server. Pilih yes.

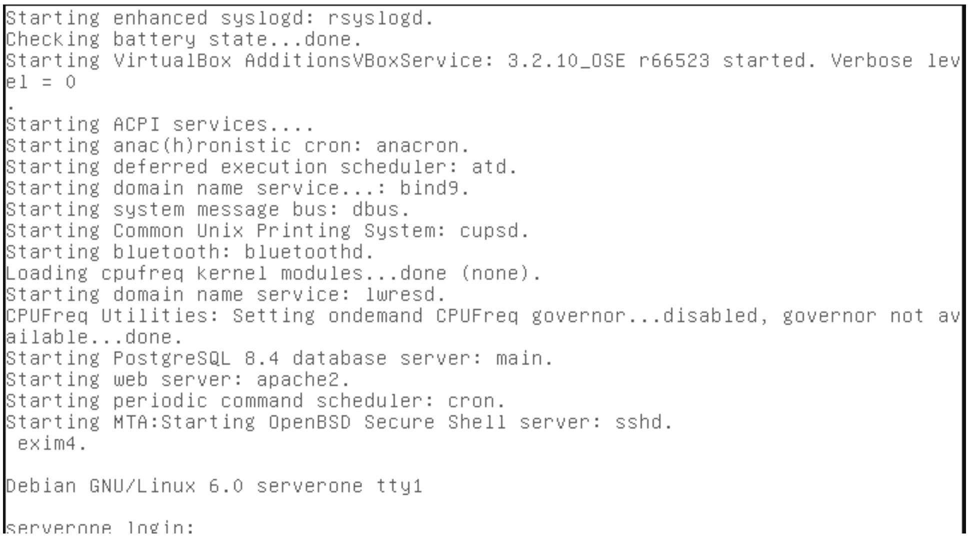
u. Selesai!



Dengan adanya pesan di atas, maka proses instalasi Debian server sudah selesai.

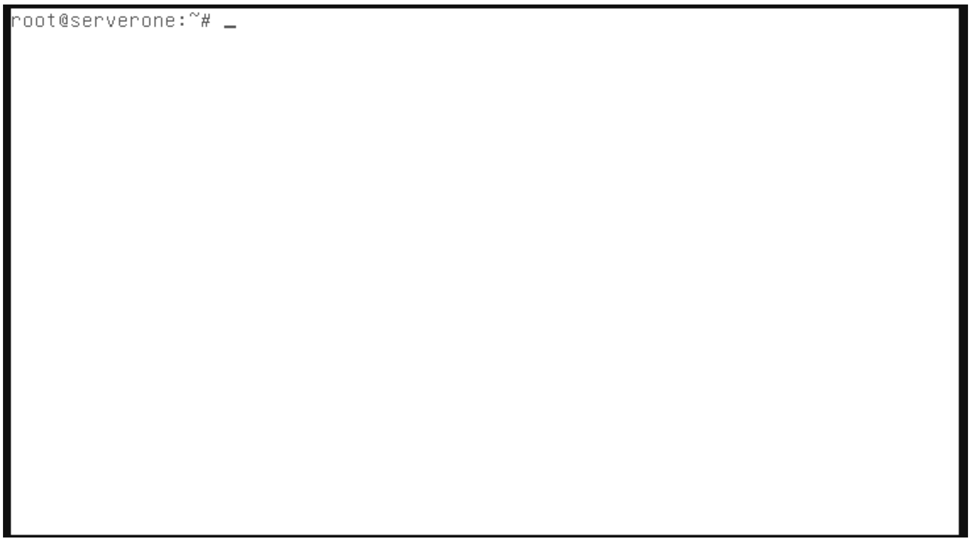
7. **Operasi Dasar Linux Debian**

Karena kita menginstall server Debian tanpa grafik. Maka hanya akan ada tampilan command line. Bagi pengguna Windows mungkin tidak terbiasa menggunakan tampilan command line. Tapi, dengan sedikit latihan pasti terbiasa.



Setelah instalasi, maka Debian akan berjalan sendiri. Tampilan pertama adalah sistem meminta kita untuk masuk ke dalam sistem. Kita telah membuat 2 akun, akun utama (root), dan akun dengan nama depan tadi.

Masuk menggunakan root, gunakan **root** sebagai serverone login, dan **root** sebagai serverone password.



Setelah masuk sebagai root, ketikkan **clear**.

Tampilan di atas adalah saat kita sudah masuk sebagai root. Tanda apabila kita sedang berada di otoritas root adalah, adanya tanda **#** pagar. Ketikkan **help** untuk melihat daftar perintah yang bisa diketikkan.

a. Rangkuman

Melakukan instalasi sistem operasi server dengan Linux Debian cukup singkat dan mudah. Dibutuhkan perangkat dasar seperti PC yang akan dijadikan server dan master dari OS Debian.

Setelah OS Debian dan PC tersiapkan, masukkan DVD Debian ke dalam DVD-ROM PC, dan boot melalui DVD-ROOM PC. Setelah masuk, maka prosesnya cukup menjelaskan sendiri.

Seperti memilih bahasa installasi dan sistem, memilih locale, pengaturan waktu, pengaturan partisi, memilih paket yang diinstal, juga mengatur pengguna dari sistem tersebut. Semua proses dilakukan secara bertahap, hingga akhirnya sistem Debian sudah terinstal.

Setelah terinstal, akses informasi tertinggi berada pada akun utama (root), masuk menggunakan root dan sistem server telah berjalan.

b. Tugas

1. Siswa ditugaskan membuat kelompok yang terdiri dari 3 – 5 orang.
2. Siswa ditugaskan melakukan praktikum installasi Linux Debian.
3. Siswa ditugaskan mencatat kegiatan installasi Linux Debian dan membuat tutorial installasi.
4. Siswa ditugaskan membuat sebuah makalah perbandingan sistem operasi server (tidak boleh sama antar kelompok) dengan Debian (misal, Windows dengan Debian, Mac dengan Debian, Solaris dengan Debian, dst).
5. Siswa ditugaskan menjadikan makalah tersebut presentasi yang kompak, jelas, padat, dan singkat.

c. Tes Formatif

d. Lembar Jawaban Tes Formatif

e. Lembar Kerja Siswa

9. Kegiatan Belajar 9 : Memahami Administrasi Sistem File dan User Access Pada Linux

a. Tujuan Pembelajaran

Setelah mengikuti kegiatan belajar 9 ini, siswa diharapkan dapat:

h. Memahami adminsitrasi sistem file dan user access pada Linux.

i. **Uraian Materi**

j. **Memahami Administrasi User pada Linux.**

8. Administrasi User

Level administrasi tertinggi ada pada akun utama sebagai super user, yaitu root. Root memiliki kemampuan untuk membuat user baru, ataupun menghapus user yang sudah ada.

```
root@serverone:~# adduser bill
Adding user `bill' ...
Adding new group `bill' (1001) ...
Adding new user `bill' (1001) with group `bill' ...
Creating home directory `/home/bill' ...
Copying files from `/etc/skel' ...
Enter new UNIX password:
Retype new UNIX password:
passwd: password updated successfully
Changing the user information for bill
Enter the new value, or press ENTER for the default
        Full Name []: Bill Gates
        Room Number []:
        Work Phone []:
        Home Phone []:
        Other []:
Is the information correct? [Y/n]
root@serverone:~# _
```

Perintah **adduser** digunakan untuk menambahkan user baru. Caranya dengan mengetik **adduser <nama_pengguna>** , maka akan membuat user baru dengan nama yang sudah dimasukkan.

Setelah nama pengguna dimasukkan, maka diminta untuk memasukkan kata sandi untuk pengguna tersebut, dan informasi yang berkaitan denganya.

Ketikan **logout** untuk keluar dari sesi root. Lalu masuk sebagai pengguna yang baru saja dibuat, dalam kasus ini nama penggunanya adalah **bill** dengan kata sandi **gates**.

```
root@serverone:~# logout

Debian GNU/Linux 6.0 serverone tty1

serverone login: bill
Password:
Linux serverone 2.6.32-5-686 #1 SMP Mon Sep 23 23:00:18 UTC 2013 i686

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
bill@serverone:~$ _
```

Apakah terlihat perbedaan sebelumnya? Iya, perbedaanya terletak pada bagian prompt dari command line.

Sebelumnya, saat kita masuk sebagai **root**, prompt dari command line berbentuk

```
root@serverone:~#
```

Namun saat kita sudah berganti user, dan masuk sebagai bill, maka prompt dari command line berbentuk

```
bill@serverone:~$
```

Tanda **~** atau tilde, menunjukkan posisi direktori yang sedang aktif. Dalam kasus ini bill tidak aktif didirektori manapun.

Sekarang, coba logout dan masuk lagi sebagai root. Ketikkan perintah berikut,




```
Debian GNU/Linux 6.0 serverone tty1

serverone login: root
Password:
Last login: Fri Dec 13 14:00:44 WIT 2013 on tty1
Linux serverone 2.6.32-5-686 #1 SMP Mon Sep 23 23:00:18 UTC 2013 i686

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
root@serverone:~# userdel -r bill
root@serverone:~# _
```

Keluar dari bill, masuk lagi menjadi root, dan menggunakan perintah **userdel –r <nama_pengguna>** untuk menghapus semua data di direktori **/home** pengguna dan juga pengguna tersebut.

9. **Administrasi Group**

   Beberapa user yang memiliki hak akses yang sama, bisa dimasukkan kedalam sebuah group. Group sangat berfungsi sehingga kita tidak perlu membatasi hak akses terhadap user satu per satu.

Untuk membuat group, ketikkan perintah berikut.

```
root@serverone:~# groupadd scoutlegion
root@serverone:~# groupdel scoutlegion
root@serverone:~# _
```

Perintah **groupadd <nama_group>** digunakan untuk membuat group, sedangkan **groupdel <nama_group>** digunakan untuk menghapus group yang ada.

Buat dua buah group, dan buat tiga user baru.

```
root@serverone:~# groupadd scoutlegion
root@serverone:~# groupadd millitary
root@serverone:~# useradd eren
root@serverone:~# useradd levi
root@serverone:~# useradd brian
root@serverone:~# _
```

Setelah itu, gunakan perintah

```
usermod –g <nama_group> <nama_user>
```

Perintah di atas digunakan untuk memindahkan user ke suatu group. Pada kasus ini, **eren** dan **levi** dipindahkan ke dalam **scoutlegion**.

```
root@serverone:~# usermod -g scoutlegion eren
root@serverone:~# usermod -g scoutlegion levi
root@serverone:~# _
```

Pindahkan **brian** dipindahkan ke dalam **millitary**.

10. **Administrasi File**

Linux tidak memiliki sistem drive seperti Windows. Hanya ada direktori-direktori, setiap user memiliki directory user sendiri-sendiri. Ada satu direktori akar, yaitu root directory, yang bisa diakses dengan menggunakan **/** slash.

Coba ketikkan **ls /**, untuk menampilkan daftar direktori dan file yang ada di root directory.

```
root@serverone:~# ls /
bin    etc          lib          mnt   root     srv  usr
boot   home         lost+found   opt   sbin     sys  var
dev    initrd.img   media        proc  selinux  tmp  vmlinuz
root@serverone:~# _
```

Perintah **ls <nama_direktori>** digunakan untuk menampilkan isi dari sebuah direktori.

Sekarang, gunakan perintah **cd /usr**, untuk mengganti direktori yang aktif menjadi direktori /usr.

```
root@serverone:~# cd /usr
root@serverone:/usr# _
```

Sekarang tanda ~ tilde berubah menjadi nama direktori yang sedang aktif.

Untuk membuat sebuah directory, gunakan perintah **mkdir <nama_directory>**.

```
root@serverone:/usr# mkdir root
root@serverone:/usr# _
```

Untuk melihat apakah directory tersebut sudah terbuat, gunakan **ls**. Apabila perintah **ls** digunakan tanpa memberikan nama direktori, maka

ls akan mendaftar semua isi direktori yang sedang aktif sekarang (yaitu /usr).

File dan direktori di Linux mempunyai aturan penggunaan, tidak sembarang user bisa mengakses semua file. Root bisa mengakses semua file yang ada, root juga bisa memberikan hak akses kepada user untuk mengakses file tertentu.

Buat sebuah user dengan nama **eren**, lalu logout dan masuk sebagainya.

Untuk membuat sebuah file kosong, gunakan perintah **touch <nama_file>**.

Lalu, masuk kembali ke dalam direktori /usr.

Ketikkan **ls –l**, untuk menampilkan isi direktori beserta attribut-attributnya,

```
eren@serverone:/usr$ ls -l
total 80
drwxr-xr-x   2 root root  20480 Dec 13 10:43 bin
drwxr-xr-x   2 root root   4096 Sep 24  2012 games
drwxr-xr-x  32 root root   4096 Dec 13 10:36 include
drwxr-xr-x  75 root root  20480 Dec 13 10:44 lib
drwxr-xr-x   3 root root   4096 Dec 13 10:16 lib64
drwxrwsr-x  10 root staff  4096 Dec 13 09:57 local
drwxr-xr-x   2 root root   4096 Dec 13 14:25 root
drwxr-xr-x   2 root root  12288 Dec 13 10:47 sbin
drwxr-xr-x 118 root root   4096 Dec 13 10:43 share
drwxr-xr-x   5 root root   4096 Dec 13 10:17 src
eren@serverone:/usr$ _
```

Directory root yang baru saja kita buat dimiliki oleh user bernama **root**. Di kolom pertama, ada penjelasan tentang hak akses dari file atau direktori tersebut.

drwxr-xr-x

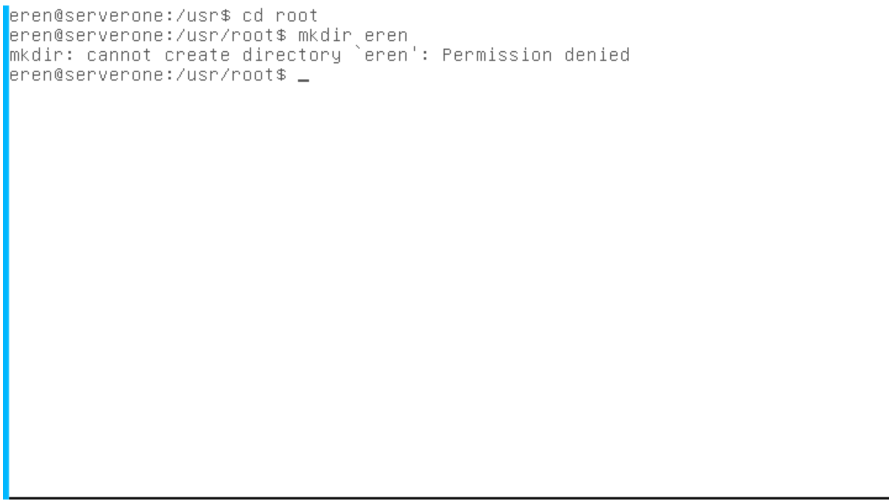
Terdiri dari 10 karakter,

1. Huruf **d** apabila direktori, **-** apabila file.



2. Hak akses untuk pemilik file,
   a. **r** – Membaca atau **–** untuk menghilangkan hak membaca
   b. **w** – Menulis atau **–** untuk menghilangkan hak menulis
   c. **x** – Mengeksekusi atau **–** untuk menghilangkan hak mengeksekusi.
3. Hak akses untuk group pemilik file,
   a. **r** – Membaca atau **–** untuk menghilangkan hak membaca
   b. **w** – Menulis atau **–** untuk menghilangkan hak menulis
   c. **x** – Mengeksekusi atau **–** untuk menghilangkan hak mengeksekusi.
4. Hak akses untuk selain pemilik dan group pemilik,
   a. **r** – Membaca atau **–** untuk menghilangkan hak membaca
   b. **w** – Menulis atau **–** untuk menghilangkan hak menulis
   c. **x** – Mengeksekusi atau **–** untuk menghilangkan hak mengeksekusi.

Sekarang, coba untuk masuk ke dalam direktori **root**, dan buatlah sebuah direktori di sana.



Karena kita aktif sebagai **eren**, dia dianggap sebagai pihak yang tidak memiliki file dan juga tidak termasuk group pemilik file. Berdasarkan aturan permission, maka kita hanya bisa membaca dan mengeksekusi dari direktori **root**.

Silahkan logout lagi, masuk sebagai root. Masuk ke direktori **/usr**, ketikkan

```
chmod o+w root
```

```
root@serverone:/usr# chmod o+w root
root@serverone:/usr# ls -l
total 80
drwxr-xr-x   2 root root  20480 Dec 13 10:43 bin
drwxr-xr-x   2 root root   4096 Sep 24  2012 games
drwxr-xr-x  32 root root   4096 Dec 13 10:36 include
drwxr-xr-x  75 root root  20480 Dec 13 10:44 lib
drwxr-xr-x   3 root root   4096 Dec 13 10:16 lib64
drwxrwsr-x  10 root staff  4096 Dec 13 09:57 local
drwxr-xrwx   2 root root   4096 Dec 13 14:25 root
drwxr-xr-x   2 root root  12288 Dec 13 10:47 sbin
drwxr-xr-x 118 root root   4096 Dec 13 10:43 share
drwxr-xr-x   5 root root   4096 Dec 13 10:17 src
root@serverone:/usr# _
```

Perintah **chmod** digunakan untuk merubah hak akses terhadap suatu file atau direktori. Dalam hal ini, kita merubah hak akses untuk direktori **root** sehingga bisa diakses oleh user lainya.

Manual dari penggunaan **chmod** mengatakan aturan untuk pemberian hak akses,

```
[ogua][[+-=][rwxXst]]
```

Berarti, di bagian pertama bisa ada huruf **o g u** dan **a.**

a. Huruf o, menyatakan hak akses untuk other, yaitu user selain group dan user pembuat.
b. Huruf g, menyatakan hak akses untuk group.
c. Huruf u, menyatakan hak akses untuk pemilik.
d. Huruf a, menyatakan hak akses semua.

Lalu diikuti dengan huruf **+ -** atau **=**, huruf **+** akan menambahkan aturan hak akses, huruf **–** akan mengurangi, huruf **=** akan membuat baru.



Lalu diikuti dengan huruf **r w x X s** dan **t**.

1. Huruf r, menyatakan hak membaca.
2. Huruf w, menyatakan hak menulis.
3. Huruf x, menyatakan eksekusi.

Untuk mencoba apakah benar user lainya sudah bisa menulis di direktori **/usr/root**, coba login kembali sebagai eren dan coba buat folder eren di dalam direktori **/usr/root**.

```
eren@serverone:/usr/root$ mkdir eren
eren@serverone:/usr/root$ ls -l
total 4
drwxr-xr-x 2 eren eren 4096 Dec 13 14:56 eren
eren@serverone:/usr/root$ _
```

Tidak ada lagi pesan bahwa aksi kita ditolak, kita bisa membuat direktori eren di dalam direktori **/usr/root** dan saat kita melihat permissionya, terlihat bahwa pemilik file tersebut adalah eren.

Untuk memindahkan file tersebut ke tangan yang lain, coba kita masuk sebagai root, masuk ke direktori /usr. Buat direktori **scoutlegion** dan **millitary**.

```
root@serverone:~# cd /usr
root@serverone:/usr# mkdir scoutlegion
root@serverone:/usr# chown eren:scoutlegion scoutlegion
root@serverone:/usr# ls -l
total 84
drwxr-xr-x   2 root root        20480 Dec 13 10:43 bin
drwxr-xr-x   2 root root         4096 Sep 24  2012 games
drwxr-xr-x  32 root root         4096 Dec 13 10:36 include
drwxr-xr-x  75 root root        20480 Dec 13 10:44 lib
drwxr-xr-x   3 root root         4096 Dec 13 10:16 lib64
drwxrwsr-x  10 root staff        4096 Dec 13 09:57 local
drwxr-xrwx   3 root root         4096 Dec 13 14:56 root
drwxr-xr-x   2 root root        12288 Dec 13 10:47 sbin
drwxr-xr-x   2 eren scoutlegion  4096 Dec 13 15:39 scoutlegion
drwxr-xr-x 118 root root         4096 Dec 13 10:43 share
drwxr-xr-x   5 root root         4096 Dec 13 10:17 src
root@serverone:/usr# _
```



a. **Rangkuman**

Administrasi user dan file sepenuhnya bisa dilakukan oleh root. Setelah akses penuh telah diberikan, maka root mempunyai hak untuk membuat user. Linux merupakan sistem operasi multi-user, kita bisa membuat banyak user dan menggunakanya.

Root tidak hanya bisa membuat, tapi juga bisa menghapus. Apabila ada sebuah file atau direktori yang ingin dilindungi, maka root bisa membuat sebuah hak akses terhadap suatu file atau direktori.

b. **Tugas**

1. Mengapa diharuskan melakukan administrasi pengguna?
2. Apa saja yang bisa dilakukan oleh super user?
3. Sebutkan dan jelaskan cara membuat user!
4. Sebutkan dan jelaskan cara menghapus user!
5. Apa yang dimaksud dengan user group?
6. Sebutkan dan jelaskan cara membuat group!
7. Sebutkan dan jealaskan cara memasukkan anggota ke group!
8. Mengapa diharuskan melakukan administrasi hak akses?
9. Sebutkan dan jelaskan cara mengatur hak akses!
10. Sebutkan dan jelaskan cara merubah kepemilikan!

c. **Test Formatif**

1. Pengguna dengan hak akses tertinggi adalah ...
   a. Root
   b. Admin
   c. DB2Admin
   d. Core
   e. System
2. Perintah yang digunakan untuk menambahkan pengguna adalah
   a) addsystem
   b) addroot

c) adduser

d) add

e) usermod

3. Beda antara super user dengan user biasa adalah super user ditandai dengan
    a. ~
    b. #
    c. $
    d. %
    e. !
4. Perintah untuk menentukan kata sandi adalah ...
    a. password
    b. pass
    c. passwd
    d. pwd
    e. pword
5. Perintah untuk membuat group adalah ...
    a. groupadd
    b. addgroup
    c. add group
    d. group add
    e. Salah semua
6. Perintah usermod -g scoutlegion erenjaegar berfungsi untuk ...
    a. Membuat user dengan nama scoutlegion dengan group erenjaegar
    b. Membuat user dengan nama erenjaegar keluar dari group scout legion
    c. Membuat user dengan nama erenjaegar masuk ke group scoutlegion
    d. Menghapus user scoutlegion dan erenjaegar
    e. Semua salah

7. Letak direktori root di Linux adalah ...
    a. C:/
    b. D:/
    c. /
    d. /super
    e. super/
8. Perintah untuk menampilkan daftar file lengkap dengan hak aksesnya adalah
    a. ls -l
    b. ls
    c. ls -chmod
    d. ls -rule
    e. ls -force
9. Hak akses untuk pemilik bisa semua, dan group bisa membaca dan menulis, sedangkan orang lain hanya membaca adalah ...
    a. -rwxrwx-wx
    b. -rwxrw-r--
    c. -rw-rwx-wx
    d. --wx--r-rwx
    e. Semua salah
10. Perintah yang digunakan untuk membuat file adalah ...
    a. touch
    b. vim
    c. nano
    d. Semua benar
    e. Semua salah

f. Lembar Jawaban Tes Formatif

g. Lembar Kerja Siswa

10. Kegiatan Belajar 10 : Menyajikan Hasil Instalasi Sistem Operasi Server

a. Tujuan Pembelajaran :Setelah mengikuti kegiatan belajar 10 ini, siswa diharapkan dapat:

k. Melakukan instalasi sistem operasi server Linux Debian.

l. Mengoperasikan sistem operasi server Linux Debian tingkat dasar.

b. Uraian Materi

m. **Melakukan Instalasi Sistem Operasi Server Linux Debian**

11. Instalasi Debian

Pastikan DVD Debian sudah siap, dengan PC yang akan dijadikan server.

a. Masukkan DVD Debian ke DVD-ROM PC.

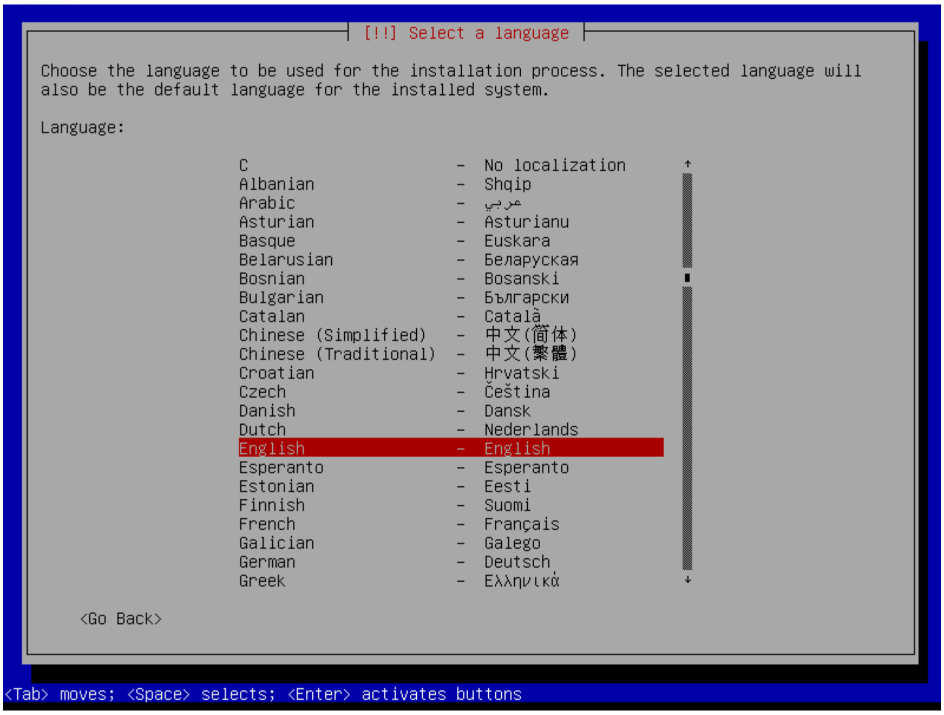
b. Masuk BIOS, pastikan PC boot ke DVD-ROM terlebih dahulu.

c. Akan muncul tampilan seperti berikut,



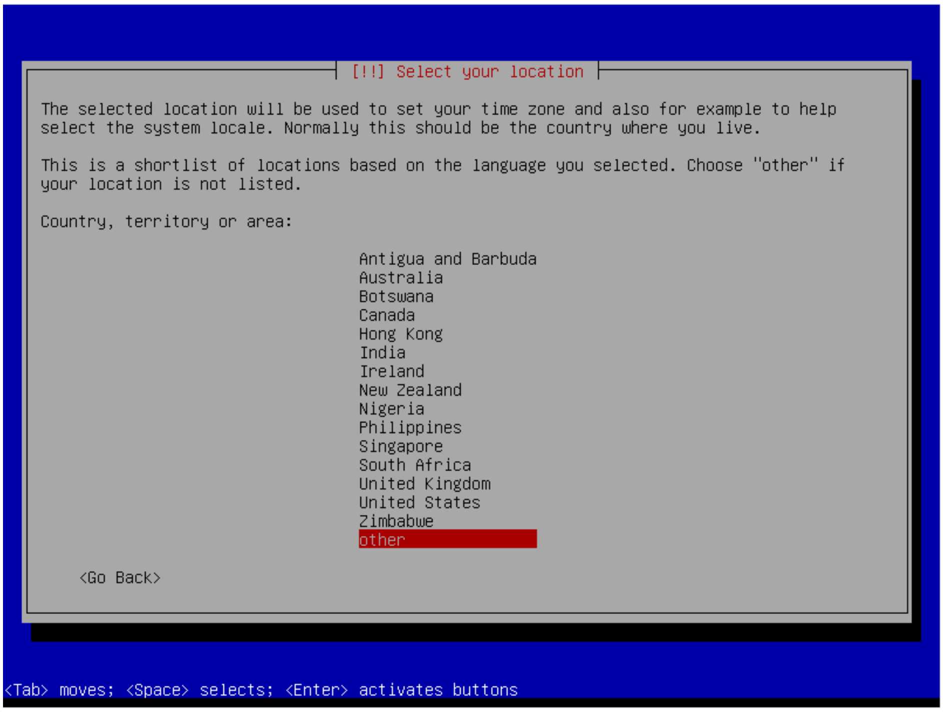
Karena kita menginstall untuk server, dan supaya proses instalasi lebih cepat, kita tidak menggunakan instalasi grafik, tapi dengan instalasi *command line*.

d. Pilih bahasa yang akan digunakan saat proses instalasi, bahasa yang dipilih juga akan menjadi bahasa yang digunakan sistem.



Pilih english, sebagai bahasa instalasi.

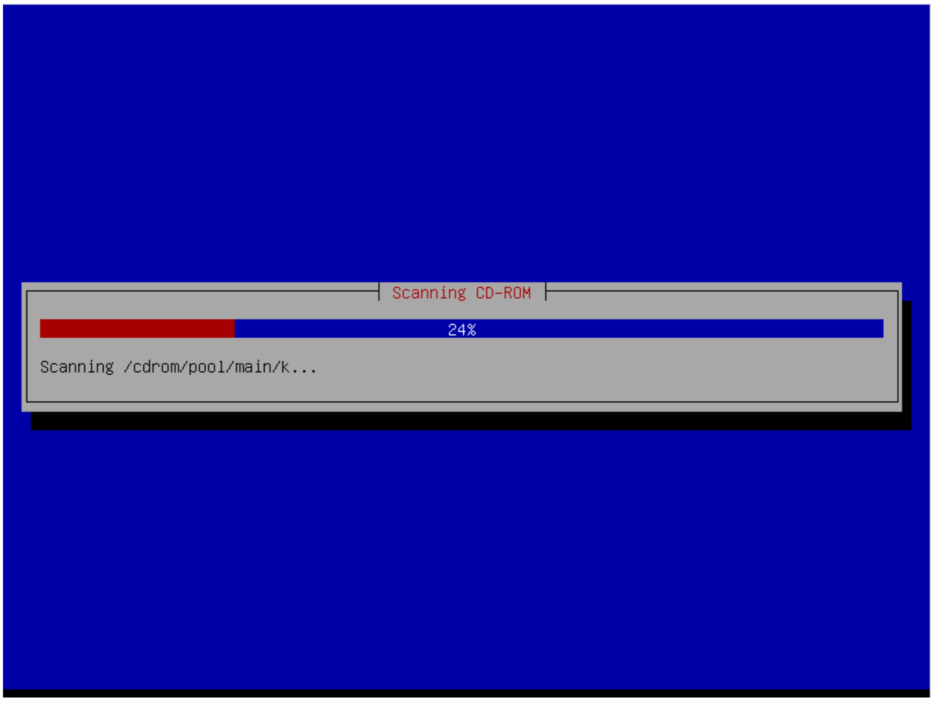
e. Tentukan lokasi negara, Indonesia berada di bagian other | Asia | Indonesia.



Pilih Indonesia sebagai negara, akan muncul pemilihan *locale* atau jenis huruf yang didukung. Indonesia menggunakan alphabet latin, gunakan locale Amerika Serikat.

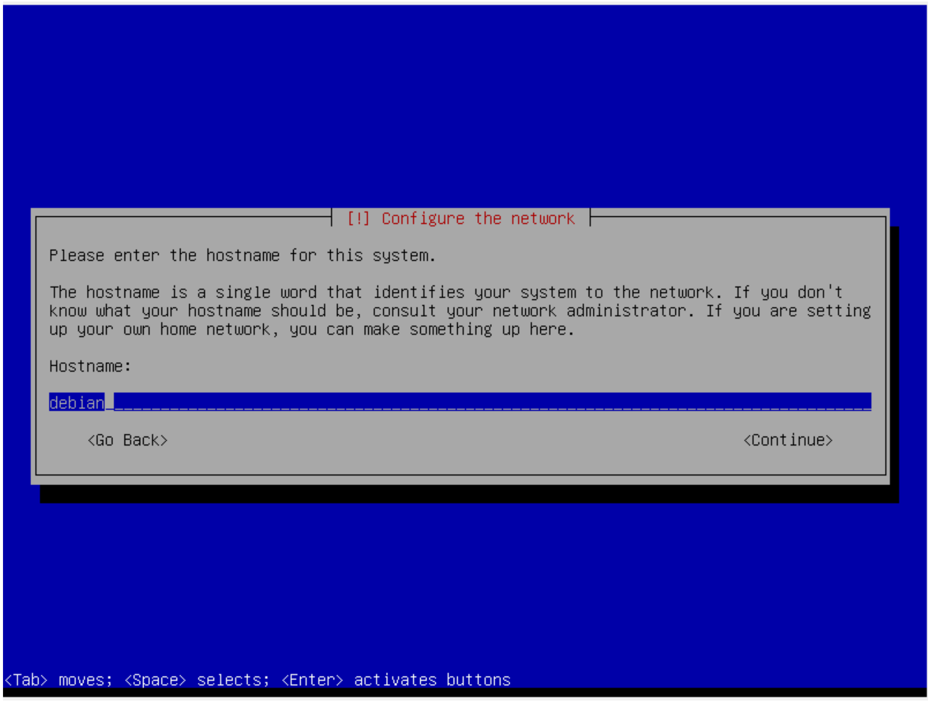
Pilih juga *keymap*, keymap adalah tatanan keyboard yang digunakan. Gunakan **American English**.

f. Tunggu, akan ada proses.



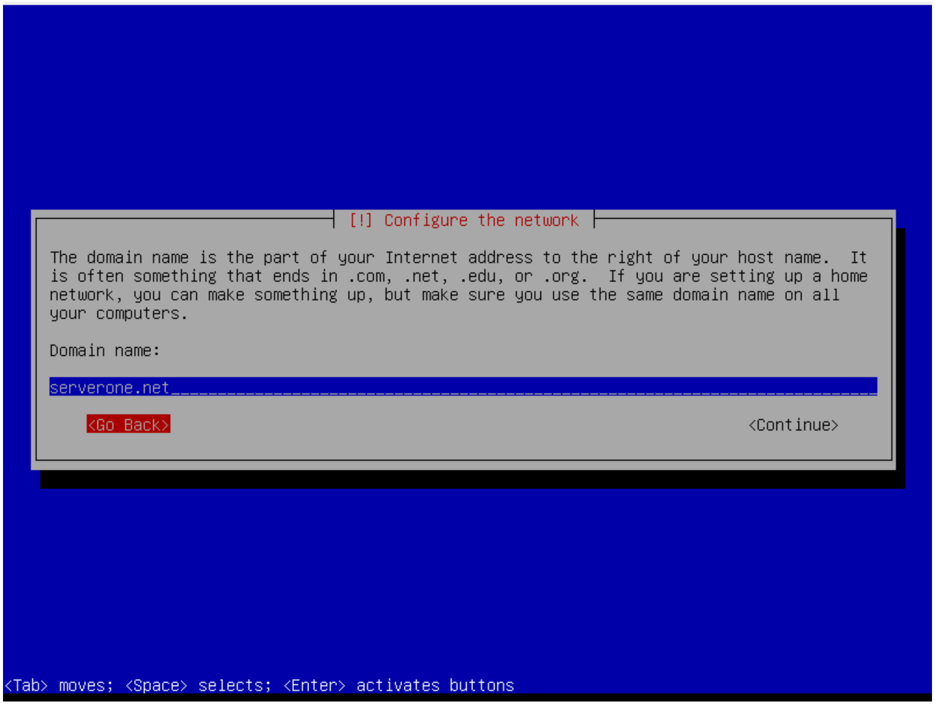
Biarkan prosesnya sampai selesai.

g. Masukkan hostname yang dibutuhkan, karena kita mensetting server kita sendiri. Masukkan sesuai keinginan.



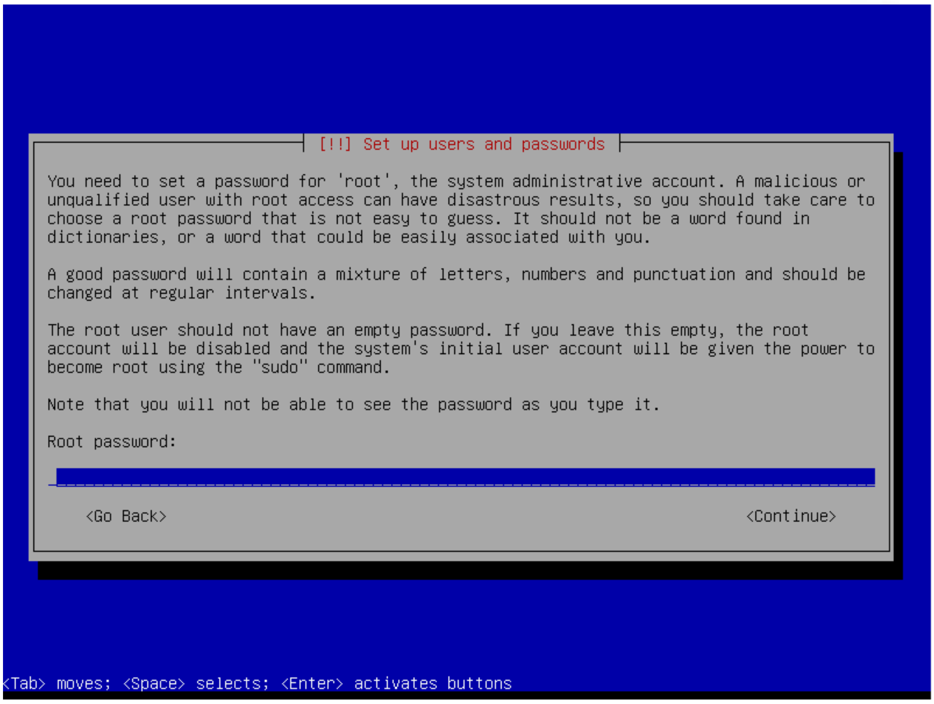
Dalam kasus ini, masukkan **serverone** sebagai hostname.

h. Setelah itu kita akan diminta memasukkan nama domain, karena kita mensetting server kita sendiri, masukan sesuai keinginan.



Dalam kasus ini, masukkan **serverone.net**.

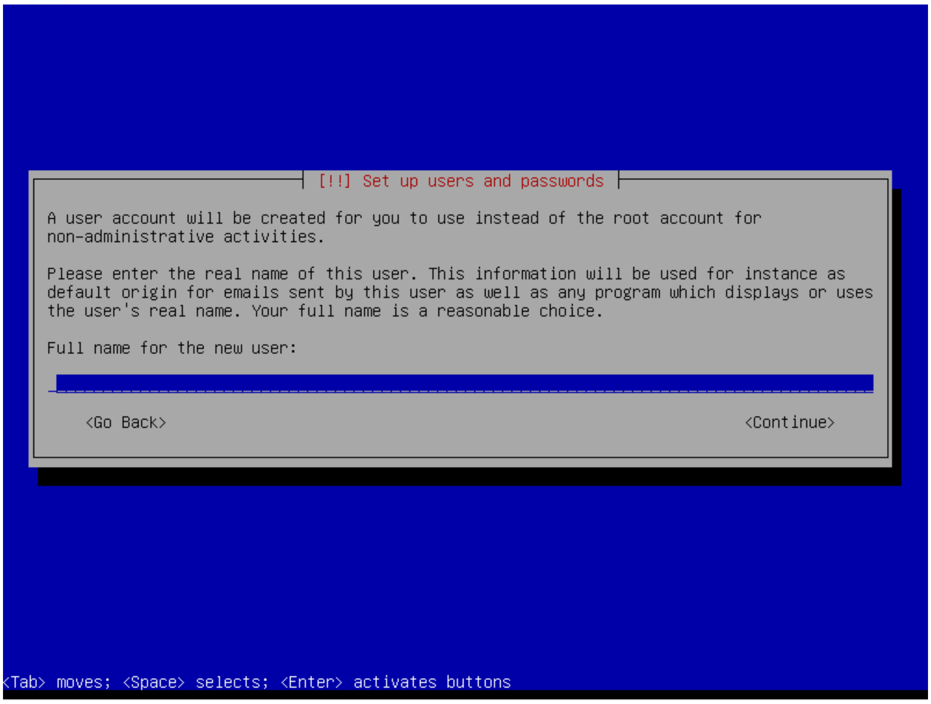
i. Kita akan memasukkan kata sandi untuk akun utama, masukkan sesuai kenginan. Kata sandi yang baik adalah kata yang tidak ada di dalam kamus, terdiri dari huruf dan angka, atau simbol-simbol tertentu, dan dirubah secara berkala.



Dalam kasus ini, masukkan **root**, lalu masukan **root** lagi saat konfirmasi sandi.

Akun utama atau root, memiliki hak akses paling tinggi. Dia bisa melakukan apa saja dengan sistem dan tidak ada yang membatasinya. Pastikan ketika anda membuat server asli, kata sandi akun root anda sangat kuat.

j. Setelah itu, akan diminta untuk memasukkan nama lengkap pengguna. Masukan nama anda.



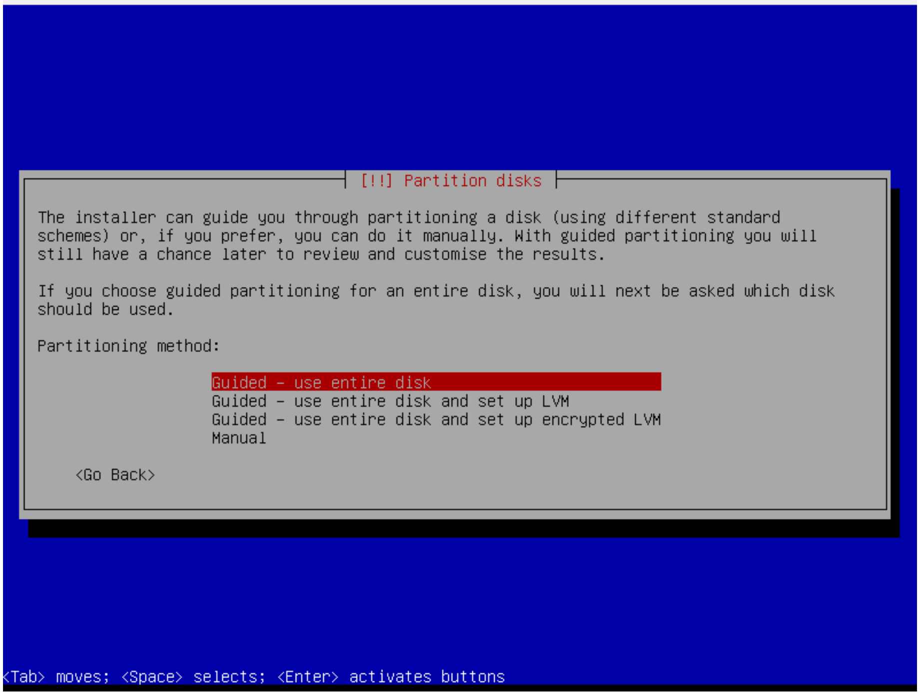
k. Masukkan nama pengguna untuk akun pribadi anda sendiri. Nama depan merupakan plihan yang cukup. Masukkan juga kata sandi, lalu konfirmasi lagi kata sandi.

l. Masukkan kota tempat anda tinggal, digunakan untuk menentukan waktu.

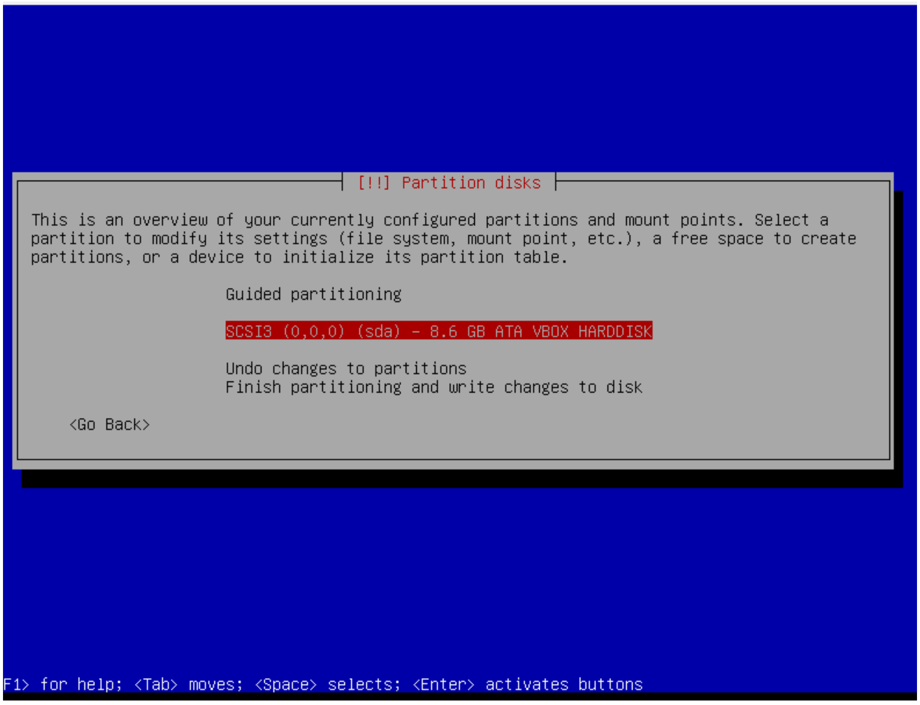


m. Tunggu sampai proses pengecekkan hardware selesai.

n. Pilih proses partisi secara manual,

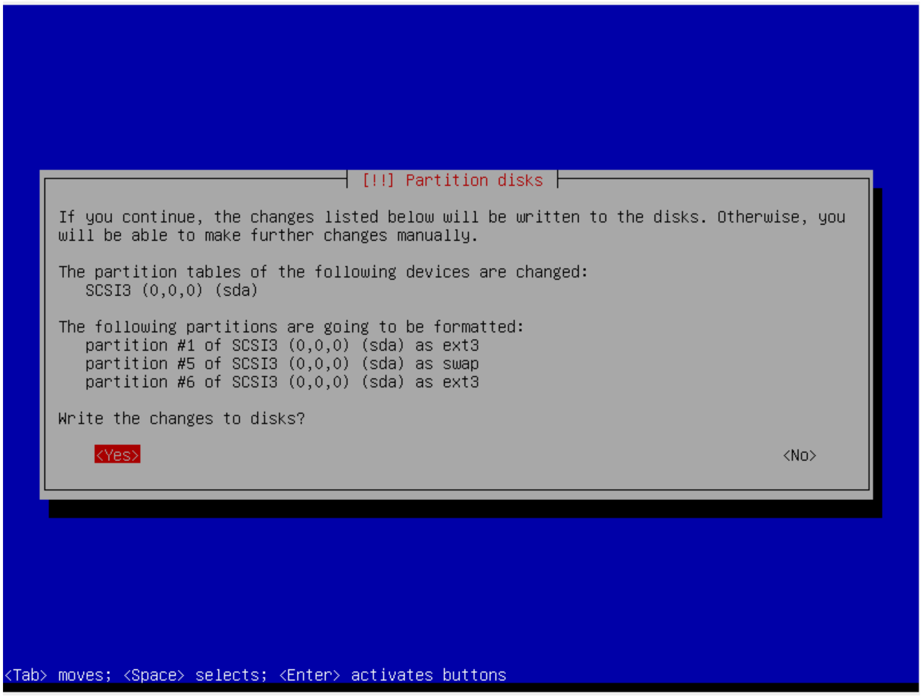


Pilih disk yang akan di partisi, masukkan 50% sebagai ukuran disk.



Tekan enter, dan pilih yes, lihat bagian bootable flag, aktifkan bagian bootable flag.

Lakukan hal yang sama pada sisa partisi lainya.

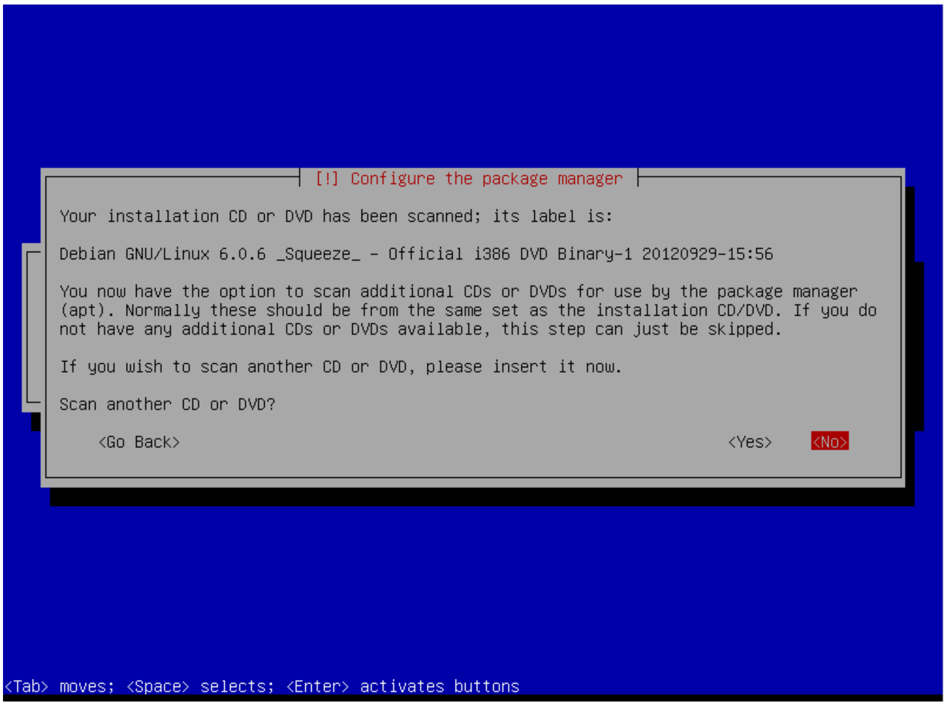


Dalam kasus ini, di server contoh menggunakan 3 partisi. Satu root, satu swap, dan satu home. Pilih yes, lalu biarkan proses partisi berjalan.

o. Biarkan proses instalasi berjalan.



p. Apabila dalam proses instalasi ada pertanyaan,



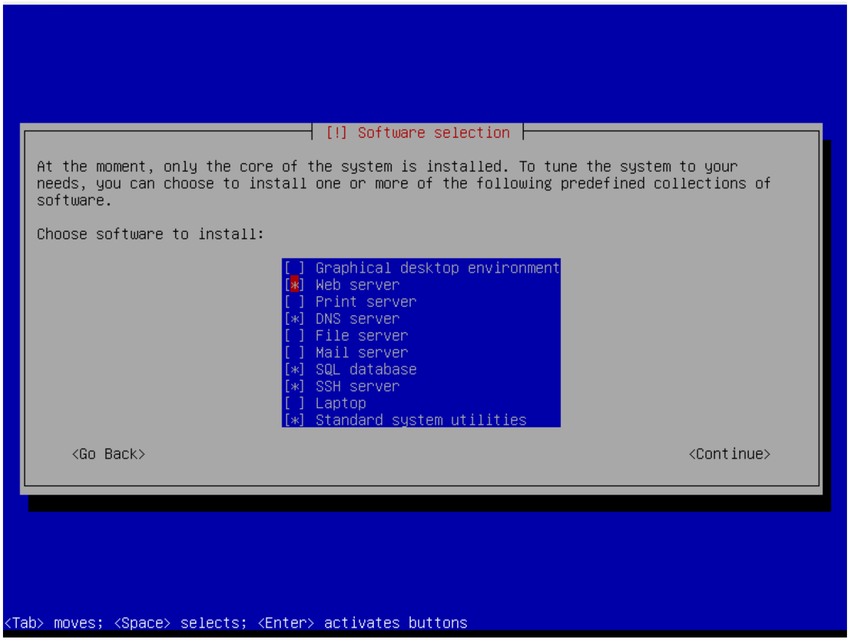
Apakah ada media instalasi lainya? DVD atau CD? Apabila ada, masukkan DVD dan pilih yes, apabila tidak, pilih no.

q. Akan ada pertanyaan lagi, apakah perlu menggunakan network mirror? Network mirror adalah master debian yang ada di internet, berfungsi untuk mengambil data-data yang mungkin tidak ada di DVD atau CD.

   Silahkan pilih ya apabila anda ingin menggunakan network mirror, syaratnya anda harus mempunyai koneksi internet.

r. Apabila ada pertanyaan seputar berkontribusi untuk popularity contest, pilih no.

s. Setelah itu akan disediakan pilihan apa saja yang akan diinstal.



Karena kita membuat sebuah server sederhana, silahkan pilih web server, DNS server, SQL Database, SSH Server, dan Standard System Utilities.

Lalu biarkan proses instalasi berjalan kembali.

t. Akan ada pesan, apabila PC yang sudah anda sudah ada sistem operasinya. Maka menginsall GRUB akan embuat sistem operasi yang sebelumnya tidak bisa digunakan. Kecuali nanti diatur ulang.

   Kita asumsikan bahwa anda belum menginstall apa-apa karena tujuanya adalah membuat PC khusus server. Pilih yes.

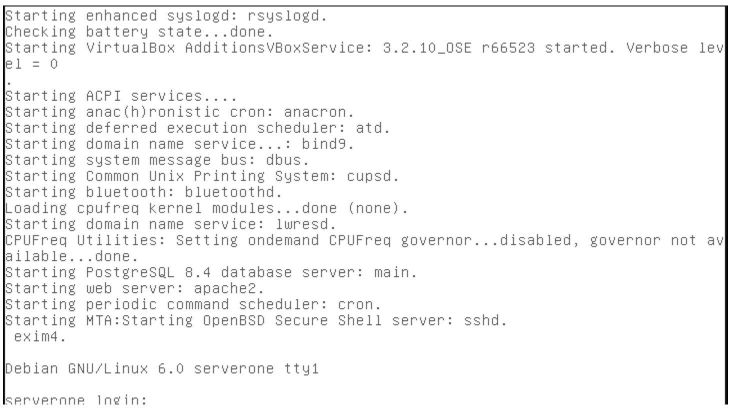
u. Selesai!



Dengan adanya pesan di atas, maka proses instalasi Debian server sudah selesai.

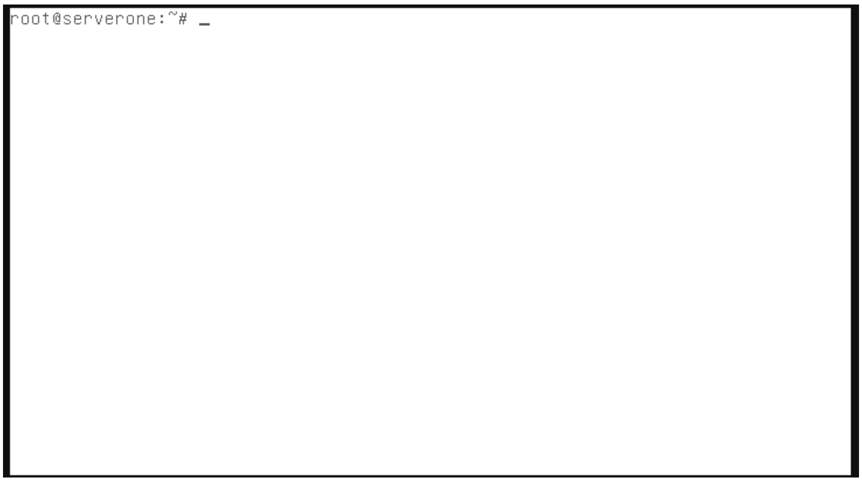
12. **Operasi Dasar Linux Debian**

Karena kita menginstall server Debian tanpa grafik. Maka hanya akan ada tampilan command line. Bagi pengguna Windows mungkin tidak terbiasa menggunakan tampilan command line. Tapi, dengan sedikit latihan pasti terbiasa.



Setelah instalasi, maka Debian akan berjalan sendiri. Tampilan pertama adalah sistem meminta kita untuk masuk ke dalam sistem. Kita telah membuat 2 akun, akun utama (root), dan akun dengan nama depan tadi.

Masuk menggunakan root, gunakan **root** sebagai serverone login, dan **root** sebagai serverone password.



Setelah masuk sebagai root, ketikkan **clear**.

Tampilan di atas adalah saat kita sudah masuk sebagai root. Tanda apabila kita sedang berada di otoritas root adalah, adanya tanda **#** pagar. Ketikkan **help** untuk melihat daftar perintah yang bisa diketikkan.

a. Rangkuman

Melakukan instalasi sistem operasi server dengan Linux Debian cukup singkat dan mudah. Dibutuhkan perangkat dasar seperti PC yang akan dijadikan server dan master dari OS Debian.

Setelah OS Debian dan PC tersiapkan, masukkan DVD Debian ke dalam DVD-ROM PC, dan boot melalui DVD-ROOM PC. Setelah masuk, maka prosesnya cukup menjelaskan sendiri.

Seperti memilih bahasa installasi dan sistem, memilih locale, pengaturan waktu, pengaturan partisi, memilih paket yang diinstal, juga mengatur pengguna dari sistem tersebut. Semua proses dilakukan secara bertahap, hingga akhirnya sistem Debian sudah terinstal.



Setelah terinstal, akses informasi tertinggi berada pada akun utama (root), masuk menggunakan root dan sistem server telah berjalan.

b. Tugas

1. Siswa ditugaskan membuat kelompok yang terdiri dari 3 – 5 orang.
2. Siswa ditugaskan untuk melakukan praktikum tentang administrasi pengguna, grup, hak akses, juga struktur direktori dan file.
3. Siswa ditugaskan untuk membuat laporan tentang cara mengatur pengguna, grup, hak akses, dan struktur direktori dan file dalam bentuk laporan dan apabila ada kendala dicatat dan ditulis cara penyelesaianya.
4. Siswa ditugaskan untuk membuat makalah tentang cara administrasi pengguna, grup, hak akses, dan struktur direktori dan file yang efisien.
5. Hasil makalah diolah sehingga menjadi kompak, padat, singkat, dan jelas lalu dijadikan bahan untuk presentasi.

c. Tes Formatif

d. Lembar Jawaban Tes Formatif

e. Lembar Kerja Siswa

11. Kegiatan Belajar 11 : Memahami Berbagai Layanan Jaringan

a. Tujuan Pembelajaran

Setelah mengikuti kegiatan belajar 11 ini, siswa diharapkan dapat :

Memahami berbagai layanan jaringan.

b. Uraian Materi

Layanan Jaringan.

13. *Domain Name System* (DNS) *Server*

Internet memiliki dua sistem penamaan host, yaitu *IP address* dan URL (*uniform resource locator*). Pengguna internet biasanya mengakses alamat sebuah website menggunakan URL seperti berikut ini : www.detik.com, www.yahoo.co. www.google.com dan sebagainya. Sedangkan untuk akses *IP address* jarang dipakai secara umum, karena memang susah untuk dihafalkan. Penomeran berbasis IP ini merupakan nomor unik yang hanya dimiliki oleh satu komputer yang terkoneksi di internet. Satu nomor hanya digunakan untuk satu perangkat, tetapi sebuah perangkat bisa saja memiliki banyak nomor IP.

Hubungan dari URL dan IP address ini dipetakan dengan sebuah sistem yang disebut DNS (*domain name service*). Komputer yang berperan sebagai DNS akan meneruskan permintaan kita berupa alamat URL menjadi nomor IP yang dipetakan ke URL tersebut. DNS memungkinkan para pengguna jaringan komputer menggunakan nama seperti www.filekontrol.com sebagai pengganti untuk mengganti *IP address* 192.168.1.1.

Pada saat suatu host di dalam sebuah jaringan terhubung ke jaringan lain melalui nama host maka proses ini disebut juga *fully qualified domain name* (FQDN), DNS digunakan untuk mengetahui *IP address* dari host tersebut. DNS diimplementasikan menggunakan sebuah server pusat yang mempunyai hak atas beberapa domain dan akan diarahkan ke DNS lain jika koneksi dilakukan ke domain yang di luar tanggung jawabnya.

DNS menggunakan arsitektur hirarki di dalam pemberian nama. Tingkat pertama adalah nama domain yang oleh lembaga *Internet Assigned Number Authority* (IANA) dikategorikan sebagai berikut :

- .com untuk dipakai perusahaan-perusahaan
- .edu untuk dipakai perguruan tinggi
- .gov untuk dipakai badan-badan pemerintah
- .mil untuk dipakai badan-badan militer
- .org untuk dipakai badan-badan yang tidak termasuk kategori di atas.

Selain itu untuk membedakan pemakaian nama oleh suatu negara dengan negara lain digunakan tanda misalnya .id untuk Indonesia, .au untuk Australia dan lain-lain.

Tingkat berikutnya adalah sub-domain, suatu domain dapat diterapkan ke berbagai sub-domain yang berupa bagian dari domain tersebut. Misalnya perusahaan “beta soft” mempunyai domain betasoft.com, dapat mempunyai berbagai sub-domain seperti support.betasoft.com, sales.betasoft.com.

1.1 Jenis *Name Server*

Terdapat empat jenis konfigurasi yang banyak digunakan :

a. *master* : digunakan untuk menyimpan record-record zona original dan authoritative untuk *name space* tertentu, menjawab pertanyaan dari *name server* lain yang mencari jawaban space tersebut.
b. *slave* : digunakan untuk menjawab permintaan dari *name server* lain. *Server* slave merupakan *backup* dari *server master*. *Server* ini mendapatkan informasi *name space* dari *name server master*. *Server master* akan mengirim perubahan tersebut ke *slave*nya setiap periode tertentu.
c. *caching-only* : digunakan untuk menawarkan layanan resolusi nama ke IP tetapi sama sekali tidak mengelola zona. Jawaban atas semua resolusi di-*cache* di dalam memori selama periode waktu tertentu, yang ditentukan oleh *record* zona yang diterima.
d. *forwarding* : digunakan untuk mem*forward* permintaan ke suatu *name server* untuk resolusi nama. Jika *name server* yang diminta tidak ditemukan, maka resolusi gagal.

Name server dapat berupa satu atau lebih jenis-jenis di atas. Tetapi sebagai contoh, sebuah *name server* dapat berupa *master* untuk beberapa zona, *slave* untuk zona lainnya, dan hanya menawarkan resolusi *forwarding* untuk zona tertentu.

14. *Dynamic Host Control Protocol* (DHCP) *Server*

DHCP *server* digunakan untuk memberikan *IP address* kepada *client* atau *workstation* yang memerlukan *IP address* secara otomatis. Komputer yang memberikan nomor IP disebut sebagai DHCP *server*, sedangkan komputer yang meminta nomor IP disebut sebagai DHCP *client*. Dengan demikian administrator tidak perlu lagi harus memberikan nomor IP secara manual pada saat konfigurasi TCP/IP, tapi cukup dengan memberikan referensi kepada DHCP *server*.

DHCP menggunakan 5 tahapan proses untuk memberikan konfigurasi nomor IP, antara lain :

a. *IP Least Request*

Merupakan proses saat *client* meminta nomor *IP ke server* (*broadcast* mencari DHCP *server*). Pada saat DHCP *client* dihidupkan, maka komputer tersebut melakukan *request* ke DHCP *server* untuk mendapatkan nomor IP.

b. *IP Least Offer*

DHCP menjawab dengan memberikan nomor IP yang ada di *database* DHCP. DHCP *server* (bisa satu atau lebih *server* jika memang ada) yang mempunyai no IP memberikan penawaran ke *client* tersebut.

c. *IP Lease Selection*

Client memilih penawaran DHCP *server* yang pertama diterima dan kembali melakukan *broadcast* dengan *message* menyetujui peminjaman tersebut kepada DHCP *Server*.

d. *IP Lease Acknowledge*

DHCP *server* memberikan jawaban atas pesan tersebut berupa konfirmasi no IP dan informasi lain kepada *client* dengan sebuah *ACKnowledgment*. Kemudian *client* melakukan inisialisasi dengan mengikat (*binding*) nomor IP tersebut dan *client* dapat bekerja pada jaringan tersebut. Nomor IP diberikan bersama dengan subnet mask dan default gateway. Setelah *server* memberikan nomor IP, maka *server* meminjamkan (*lease*) nomor IP

yang ada ke DHCP *client* dan mencoret nomor IP tersebut dari daftar *pool*. Jika tidak ada lagi nomor IP yang dapat diberikan, maka client tidak dapat menginisialisasi TCP/IP, dengan sendirinya tidak dapat tersambung pada jaringan tersebut.s

e. **Lease Period**

Setelah periode waktu tertentu, maka pemakaian DHCP *client* tersebut dinyatakan selesai dan *client* tidak memperbaharui permintaan kembali, maka nomor IP tersebut dikembalikan kepada DHCP *server*, dan *server* dapat memberikan nomor IP tersebut kepada *client* yang membutuhkan. Lama periode ini dapat ditentukan dalam menit, jam, bulan atau selamanya.

15. *Firewall* dan *Network Address Translation* (NAT)

3.1 Firewall

IPTABLES adalah suatu tools dalam sistem operasi linux yang berfungsi sebagai alat untuk melakukan penyaringan terhadap lalu lintas data. Secara sederhana digambarkan sebagai pengatur lalu lintas data. Dengan IPTABLES inilah kita akan mengatur semua lalu lintas dalam komputer, baik yang masuk ke komputer, keluar dari komputer, ataupun lalu lintas data yang sekedar melewati komputer.

IPTABLES fapat digunakan untuk melakukan seleksi terhadap paket-paket yang datang baik *input, output* maupun *forward* berdasarkan *IP address*, identitas jaringan, port, *source* (asal), *destination* (tujuan), protokol yang digunakan bahkan berdasarkan tipe koneksi terhadap setiap paket (data yang diinginkan).

IPTABLES dapat melakukan perhitungan terhadap paket dan menerapkan prioritas trafik berdasar jenis layanan (*service*). IPTABLES dapat digunakan untuk mendefinisikan sekumpulan aturan keamanan berbasis *port* untuk mengamankan *host-host* tertentu. IPTABLES juga dapat dimanfaatkan untuk membangun sebuah *router* atau *gateway*, tentunya hanya untuk sistem operasi Linux.

Firewall IPTABLES *packet filtering* memiliki tiga aturan (*policy*), yaitu:

a. INPUT

Mengatur paket data yang memasuki firewall dari arah intranet maupun

internet. kita bisa mengelola komputer mana saja yang bisa mengakses firewall, misal: hanya komputer IP 192.168.1.100 yang bisa mengakses SSH ke firewall dan yang lain tidak boleh.

b. OUTPUT

Mengatur paket data yang keluar dari firewall ke arah intranet maupun internet. Biasanya output tidak diset, karena bisa membatasi kemampuan firewall itu sendiri.

c. FORWARD

Mengatur paket data yang melintasi firewall dari arah internet ke intranet maupun sebaliknya. Aturan *forward* paling banyak dipakai saat ini untuk mengatur koneksi internet berdasarkan port, *mac address* dan alamat IP. Selain aturan (*policy*), firewall IPTABLES juga mempunyai parameter yang disebut dengan TARGET, yaitu status yang menentukan koneksi di IPTABLES diizinkan lewat atau tidak. TARGET ada tiga macam yaitu:

- ACCEPT

  Akses diterima dan diizinkan melewati firewall.

- REJECT

  Akses ditolak, koneksi dari komputer klien yang melewati firewall langsung terputus, biasanya terdapat pesan “Connection Refused”. Target Reject tidak menghabiskan bandwidth internet karena akses langsung ditolak, hal ini berbeda dengan DROP.

- DROP

  Akses diterima tetapi paket data langsung dibuang oleh kernel, sehingga pengguna tidak mengetahui kalau koneksinya dibatasi oleh firewall, pengguna melihat seakan – akan server yang dihubungi mengalami permasalahan teknis. Pada koneksi internet yang sibuk dengan trafik tinggi, Target Drop sebaiknya jangan digunakan.

3.2 *Network Address Translation* (NAT)

Pada jaringan komputer, proses *Network Address Translation* (NAT) adalah proses penulisan ulang (*masquerade*) pada alamat IP asal (*source*) dan/atau alamat IP tujuan (*destination*), setelah melalui *router* atau *firewall*. NAT digunakan pada jaringan dengan *workstation* yang menggunakan IP *private* supaya dapat terkoneksi ke Internet dengan menggunakan satu atau lebih IP *public*. Pada mesin Linux, untuk

membangun NAT dapat dilakukan dengan menggunakan IPTABLES (Netfilter). Dimana pada IPTABLES memiliki tabel yang mengatur NAT.

NAT dapat dikerjakan oleh kernel Linux dengan salah satu dari dua cara berikut :

a. *Source* NAT

SNAT digunakan untuk menyembunyikan asala paket-paket dengan melakukan pemetaan alamat asal paket-paket yang akan menuju jaringan eksternal ke suatu IP *address* atau *range address* tertentu. Dengan kemampuan seperti ini, SNAT bisa digunakan sebagai server Masquerader.

b. *Destination* NAT

DNAT sering digunakan untuk me-*redirect* secara transparan paket-paket yang masuk ke suatu lokasi/tujuan, misalnya diarahkan ke mesin yang berfungsi sebagai *server proxy* atau firewall SOCKS.

Salah satu versi dari NAT adalah IP Masquerade, yang mengijinkan beberapa *workstation* atau *host* terkoneksi ke internet tanpa harus memiliki IP *address* yang dapat dikenal di jaringan eksternal internet. Server yang memiliki fungsi sebagai *gateway* menyediakan suatu masquerader menggunakan IPTABLES untuk membuat *host-host* lokal dikenal di jaringan internet dimana IP *address* yang tercatat adalah IP *address gateway*.

Proses masquerade IP dikerjakan menggunakan subsitusi IP *address* dan nomor port. IP *address* paket dari jaringan lokal diubah berdasarkan pada tujuannya. Berikut ini adlaah aturan sederhananya :

a. Paket yang menuju jaringan eksternal (meninggalkan jaringan lokal menuju ke *gateway*). IP *address* asal paket diubah ke IP *address* mesin maquerader. IP *address* masquerader bersifat unik pada jaringan eksternal.

b. Paket yang masuk dari jaringan eksternal (menuju jaringan lokal melalui *gateway*). Alamat paket diubah ke IP *address host* jaringan lokal. Mesin-mesin di dalam jaringan lokal memiliki alamat *private network* yang tidak valid (tidak dikenal) pada jaringan eksternal.

IP Masquerade menggunakan *port forwarding* untuk mengubah suatu IP *address* paket. Pada saat sebuah paket sampai dari jaringan eksternal,

alamat portnya diperiksa dan dibandingkan terhadap isi tabel masquerade. Jika port yang dibandingkan ditemukan, IP *address* yang ada pada *header* paket diubah dan dikirim ke IP *address* yang telah di-masquerade.

Ada 3 hal yang harus diperhatikan dalam implementasi NAT :

- Semua aturan penterjemahan *address* ke *chain-chain* dalma tabel NAT.
- Tabel NAT menggunakan *chain-chain* seperti berikut ini :

a) PREROUTING, digunakan untuk memilah paket yang akan diteruskan

b) POSTROUTING, digunakan untuk memilah paket yang telah diteruskan

c) FORWARD, digunakan untuk memilih paket yang melalui router.

- Memasukkan modul-modul kernel untuk menangani protokol-protokol khusus.

16. Proxy *Server*

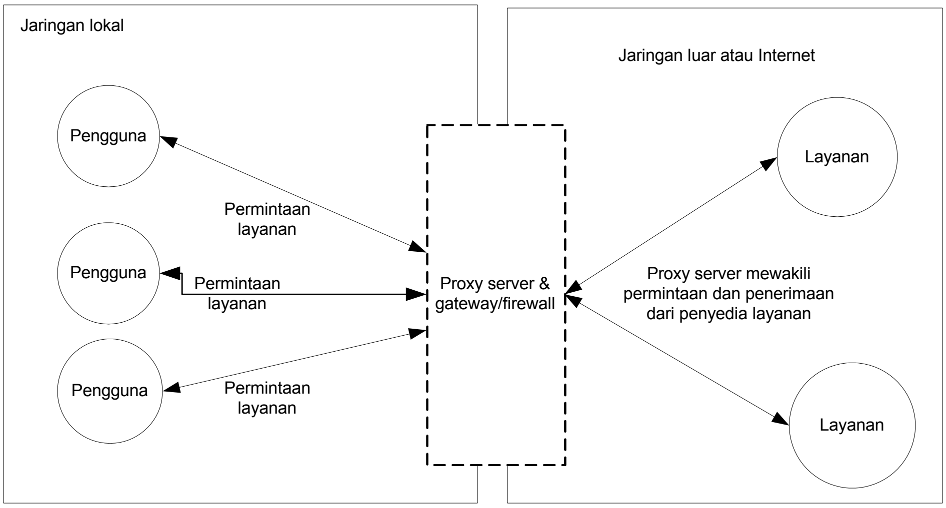
Proxy dapat dipahami sebagai pihak ketiga yang berdiri ditengah-tengah antara kedua pihak yang saling berhubungan dan berfungsi sebagai perantara, sedemikian sehingga pihak pertama dan pihak kedua tidak secara langsung berhubungan, akan tetapi masing-masing berhubungan dengan perantara, yaitu proxy.

Proxy server mempunyai 3 fungsi utama yaitu *Connection Sharing, Filtering* dan *Caching*. Proxy dalam pengertiannya sebagai perantara, bekerja dalam berbagai jenis protokol komunikasi jaringan dan dapat berada pada level-level yang berbeda pada hirarki layer protokol komunikasi jaringan. Suatu perantara dapat saja bekerja pada layer Data-Link, layer Network dan Transport, maupun layer Aplikasi dalam hirarki layer komunikasi jaringan menurut OSI. Namun pengertian proxy *server* sebagian besar adalah untuk menunjuk suatu *server* yang bekerja sebagai proxy pada layer Aplikasi, meskipun juga akan dibahas mengenai *proxy* pada level sirkuit.

Dalam suatu jaringan lokal yang terhubung ke jaringan lain atau internet, pengguna tidak langsung berhubungan dengan jaringan luar atau internet, tetapi harus melewati suatu *gateway*, yang bertindak sebagai batas antara jaringan lokal dan jaringan luar. *Gateway* ini sangat penting, karena

jaringan lokal harus dapat dilindungi dengan baik dari bahaya yang mungkin berasal dari internet, dan hal tersebut akan sulit dilakukan bila tidak ada garis batas yang jelas jaringan lokal dan internet. *Gateway* juga bertindak sebagai titik dimana sejumlah koneksi dari pengguna lokal akan terhubung kepadanya, dan suatu koneksi ke jaringan luar juga terhubung kepadanya. Dengan demikian, koneksi dari jaringan lokal ke internet akan menggunakan sambungan yang dimiliki oleh *gateway* secara bersama-sama (*connection sharing*). Dalam hal ini, *gateway* adalah juga sebagai proxy *server*, karena menyediakan layanan sebagai perantara antara jaringan lokal dan jaringan luar atau internet.

Diagram berikut menggambarkan posisi dan fungsi dari proxy *server*, diantara pengguna dan penyedia layanan:



Gambar 1 Fungsi dari proxy *server*, diantara pengguna dan penyedia layanan

4.1 Pendekatan Layer OSI

Karena proxy bekerja pada layer aplikasi, proxy *server* dapat berjalan pada banyak aplikasi antara lain HTTP Proxy atau Web Proxy untuk protokol HTTP atau Web, FTP Proxy, SMTP/POP Proxy untuk email, NNTP proxy untuk Newsgroup, RealAudio/RealVideo Proxy untuk *multimedia streaming*, IRC proxy untuk *Internet Relay Chat* (IRC), dan lain-lain. Masing-masing hanya akan menerima,meneruskan atau melakukan filter atas paket yang dihasilkan oleh layanan yang bersesuaian.

Proxy aplikasi spesifik memiliki pilihan konfigurasi yang sangat banyak. Sebagai contoh, Web Proxy dapat dikonfigurasi untuk menolak akses ke situs web tertentu pada waktu-waktu tertentu. Demikian juga proxy yang lain,

misalnya dapat dikonfigurasi untuk hanya memperbolehkan download FTP dan tidak memperbolehkan upload FTP, hanya memperbolehkan pengguna tertentu yang bisa memainkan file-file RealAudio, mencegah akses ke email server sebelum tanggal tertentu, dan masih banyak lagi.

Proxy *server* juga sangat baik dalam hal kemampuan menyimpan catatan (*logging*) dari trafik jaringan, dan dapat digunakan untuk memastikan bahwa koneksi untuk jenis trafik tertentu harus selalu tersedia. Sebagai contoh, sebuah kantor mempunyai koneksi terus menerus ke Internet untuk keperluan akses Web menggunakan satu koneksi Dial-up. Proxy *server* dapat dikonfigurasi untuk membuka satu lagi koneksi Dial-up kedua bila ada pengguna yang melakukan download melalui FTP pada koneksi Dial-up pertama dalam waktu lama.

Sebagaimana biasa, kelemahan dari konfigurasi yang sangat fleksibel dan banyak pilihan adalah timbulnya kompleksitas. Aplikasi pada sisi pengguna seperti Web Browser atau RealAudio Player harus ikut dikonfigurasi untuk bisa mengetahui adanya proxy server dan bisa menggunakan layanannya. Bila suatu layanan baru dibuat di internet yang berjalan pada layer aplikasi, dengan menggunakan protokol baru dan port yang baru, maka harus dibuat juga proxy yang spesifik dan bersesuaian dengan layanan tersebut. Proses penambahan pengguna dan pendefinisian aturan akses pada suatu proxy juga bisa sangat rumit.

Sebagai perantara antara pengguna dan server-server di internet, proxy *server* bekerja dengan cara menerima permintaan layanan dari user, dan kemudian sebagai gantinya proxy *server* akan mewakili permintaan pengguna, ke *server-server* di internet yang dimaksudkan. Dengan demikian, sebenarnya proxy *server* hanya meneruskan permintaan pengguna ke *server* yang dimaksud, akan tetapi disini identitas peminta sudah berganti, bukan lagi pengguna asal, tetapi proxy *server* tersebut. Server-server di internet hanya akan mengeahui identitas proxy *server* tersebut, sebagai yang meminta, tetapi tidak akan tahu peminta sebenarnya (yaitu pengguna asalnya) karena permintaan yang sampai kepada *server-server* di internet bukan lagi dari pengguna asal, tetapi dari proxy *server*.



Bagi penggguna sendiri, proses yang terjadi pada proxy server diatas juga tidak kelihatan (transparan). Pengguna melakukan permintaan atas layanan-layanan di internet langsung kepada *server-server* layanan di internet. Penguna hanya mengetahui keberadaan atau alamat dari proxy *server*, yang diperlukan untuk melakukan konfigurasi pada sisis pengguna untuk dapat menggunakan layanan dari proxy *server* tersebut.

4.2 *Caching*

Proxy *server* memiliki mekanisme penyimpanan obyek-obyek yang sudah pernah diminta dari *server-server* di internet, biasa disebut *caching*. Karena itu, proxy *server* yang juga melakukan proses *caching* juga biasa disebut *cache server*.

Mekanisme *caching* akan menyimpan obyek-obyek yang merupakan hasil permintaan dari dari para pengguna, yang didapat dari internet. Karena proxy *server* bertindak sebagai perantara, maka proxy *server* mendapatkan obyek-obyek tersebut lebih dahulu dari sumbernya untuk kemudian diteruskan kepada peminta yang sesungguhnya. Dalam proses tersebut, proxy *server* juga sekaligus menyimpan obyek-obyek tersebut untuk dirinya sendiri dalam ruang disk yang disediakan (*cache*).

Dengan demikian, bila suatu saat ada pengguna yang meminta suatu layanan ke internet yang mengandung obyek-obyek yang sama dengan yang sudah pernah diminta sebelumnya, yaitu yang sudah ada dalam *cache*, maka proxy *server* akan dapat langsung memberikan obyek dari cache yang diminta kepada pengguna, tanpa harus meminta ulang ke *server* aslinya di internet. Bila permintaan tersebut tidak dapat ditemukan dalam *cache* di proxy *server*, baru kemudian proxy *server* meneruskan atau memintakannya ke *server* aslinya di internet.

Proses *caching* ini juga tidak kelihatan bagi pengguna (transparan), karena bagi pengguna tidak tampak siapa sebenarnya yang yang memberikan obyek yang dimintanya, apakah proxy *server* yang mengambil dari *cache*-nya atau *server* asli di internet. Dari sisi pengguna, semua akan nampak sebagai balasan langsung dari internet.

Salah satu proxy yang paling banyak dibahas dan digunakan secara luas adalah HTTP proxy atau Web proxy. HTTP proxy *server* merupakan proxy yang berdiri diantara alokasi web pengguna misalnya web *browser* dan web *server* atau HTTP *server*. Ketika pengguna membuka browser dan mengetikkan URL, maka content yang diminta URL tersebut dinamakan “Internet Object”. Pertama dia akan bertanya terlebih dahulu ke sebuah DNS. DNS akan mencari IP Address dari URL tersebut dalam *database*nya dan memberi jawaban kepada *browser* tersebut kembali. Setelah *browser* mendapatkan IP *address*, maka ia akan membuka hubungan ke port http *web server* tujuan. *Web server* akan mendengarkan adanya permintaan dari browser lalu memberikan *content* yang diminta tersebut. Setelah browser menerima *content* maka hubungan dengan web server bias diputus. *Content* lalu ditampilkan dan disimpan didalam hardisk.

Content yang disimpan didalam hardisk biasanya disebut *cache object* yang nantinya akan digunakan jika pengguna kembali mengunjungi halaman yang sama, misalnya dengan mengklik tombol *back* atau melihat *history*. Dalam kunjungan berikutnya, browser akan memeriksa validasi *content* yang disimpannya, validasi ini dilakukan dengan membandingkan *header content* yang ada pada *cache object* dengan yang ada pada *web server*, jika *content* belum kadaluwarsa maka content tadi akan ditampilkan kembali ke browser.

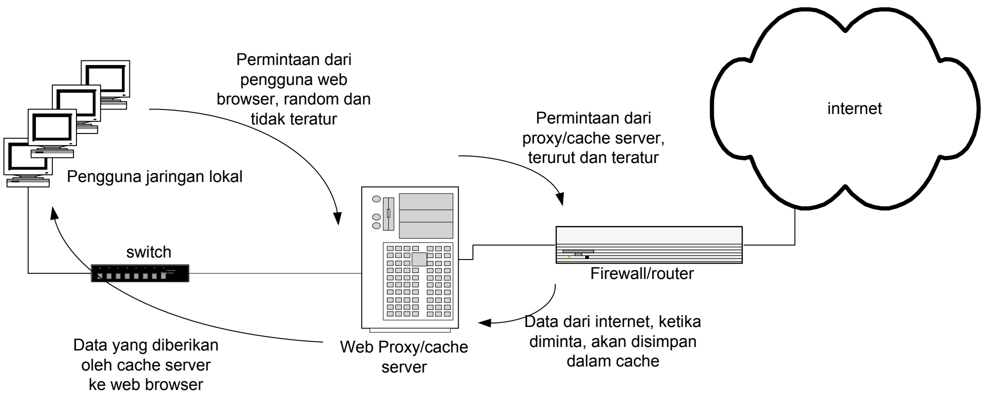
Cache object yang disimpan dalam hardisk lokal ini hanya bias dipakai oleh pengguna sendirian, tidak bisa dibagi dengan pengguna yang lainnya. Lain hal jika *content* tersebut disimpan pada sebuah *server*, dimana semua komputer terhubung dengan *server* tersebut, maka *cache object* tersebut dapat dipakai bersama-sama, server tersebutlah yang nantinya akan dinamakan *cache server* atau proxy *server*.

Cache server diletakkan pada titik diantara klien dan web server . Pada contoh diatas klien akan meminta *content* dari suatu *web server* ke *cache server*, tidak langsung ke web server tujuan. *Cache server* inilah yang bertanggung jawab untuk mendownload *content* yang diminta dan memberikannya pada klien. Content tadi disimpan pada hardisk lokal *cache server*. Lain waktu, ada klien yang meminta *content* yang sama, maka *cache server* tidak perlu mengambil langsung dari server tujuan tapi tinggal

memberikan *content* yang sudah ada. Disinilah letak optimasi *cache server* tersebut.

Ada dua jenis metode *caching*, yaitu pasif dan aktif. Seperti telah kita ketahui, object yang disimpan bisa saja mencapai kadaluarsa, untuk memeriksanya dilakukan validasi. Jika validasi ini dilakukan setelah ada permintaan dari klien, metode ini disebut pasif. Pada *caching* aktif, *cache server* mengamati *object* dan pola perubahannya. Misalkan pada sebuah *object* didapati setiap harinya berubah setiap jam 12 siang dan pengguna biasanya membacanya jam 14, maka *cache server* tanpa diminta klien akan memperbaharui *object* tersebut antara jam 12 dan 14 siang. Dengan cara *update* otomatis ini waktu yang dibutuhkan pengguna untuk mendapatkan *object* yang fresh akan semakin sedikit.

Pada kondisi tertentu, kapasitas penyimpanan akan terkuras habis oleh *object*. Namun *cache server* mempunyai beberapa metode penghapusan untuk menjaga kapasitas tetap terjaga, sesuai dengan kopnfigurasi yang telah ditetapkan. Penghapusan ini didasarkan pada umur dan kepopuleran, semakin tua umur *object* akan tinggi prioritasnya untuk dihapus. San juga untuk *object* yang tidak popular akan lebih cepat dihapus juga. Diagram berikut menggambarkan proses dan mekanisme *caching* :



Gambar 2 Proses dan mekanisme *caching*

4.3 *Transparent Proxy*

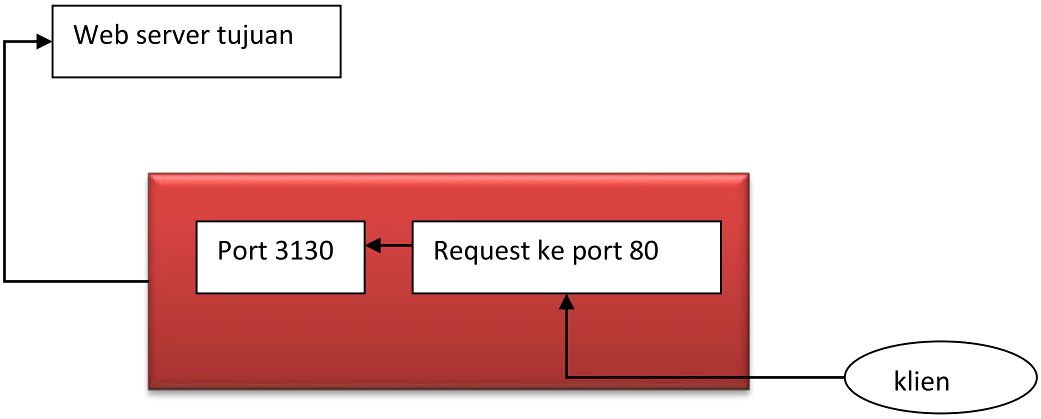
Salah satu kompleksitas dari proxy pada level aplikasi adalah bahwa pada sisi pengguna harus dilakukan konfigurasi yang spesifik untuk suatu proxy tertentu agar bisa menggunakan layanan dari suatu proxy *server*. Bila diinginkan agar pengguna tidak harus melakukan konfigurasi khusus, kita

bisa mengkonfigurasi proxy/*cache server* agar berjalan secara benar-benar transparan terhadap pengguna (*transparent* proxy). Biasanya cara ini memerlukan bantuan dan konfigurasi aplikasi firewall (yang bekerja pada layer network) untuk bisa membuat *transparent* proxy yang bekerja pada layer aplikasi.

Transparent proxy dapat berguna untuk “memaksa pengguna” menggunakan proxy/*cache server*, karena pengguna benar-benar tidak mengetahui tentang keberadaan proxy ini, dan apapun konfigurasi pada sisi pengguna, selama proxy *server* ini berada pada jalur jaringan yang pasti dilalui oleh pengguna untuk menuju ke internet, maka pengguna pasti dengan sendirinya akan “menggunakan” proxy/*cache* ini.

Cara membuat *transparent* proxy adalah dengan membelokkan arah (*redirecting*) dari paket-paket untuk suatu aplikasi tertentu, dengan menggunakan satu atau lebih aturan pada firewall/router. Hal ini bisa dilakukan karena setiap aplikasi berbasis TCP akan menggunakan salah satu port yang tersedia, dan firewall dapat diatur agar membelokkan paket yang menuju ke port layanan tertentu, ke arah port dari proxy yang bersesuaian.

Sebagai contoh, pada saat klient membuka hubungan HTTP (port 80) dengan suatu web server, firewall pada router yang menerima segera mengenali bahwa ada paket data yang berasal dari klien dengan nomor port 80. Disini kita juga mempunyai satu HTTP proxy server yang berjalan pada port 3130. Maka pada firewall router kita buat satu aturan yang menyatakan bahwa setiap paket yang datang dari jaringan lokal menuju ke port 80 harus dibelokkan ke arah alamat HTTP proxy server port 3130. Akibatnya, semua permintaan web dari pengguna akan masuk dan diwakili oleh HTTP proxy server diatas.



Gambar 3 Proses *transparent* proxy

Jadi secara umum keuntungan dari metode *transparent* proxy itu sendiri adalah :

1. Kemudahan administrasi jaringan, dengan artian browser yang digunakan *client* tidak harus dikonfigurasi secara khusus yang menyatakan bahwa mereka menggunakan fasilitas proxy yang bersangkutan.
2. Sentralisasi kontrol, dengan artian, pergantian metode *by pass* proxy maupun penggunaan proxy oleh *client* dapat dilakukan secara terpusat.

4.4 Squid Web Proxy/*Cache*

Salah satu contoh aplikasi proxy/*cache server* adalah Squid. Squid dikenal sebagai aplikasi proxy dan *cache server* yang handal. Pada pihak klien bekerja aplikasi browser yang meminta request http pada port 80. Browser ini setelah dikonfigurasi akan meminta *content*, yang selanjutnya disebut *object*, kepada *cache server*, dengan nomor port yang telah disesuaikan dengan milik *server*, nomor yang dipakai bukan port 80 melainkan port 8080 3130 (kebanyakan *cache server* menggunakan port itu sebagai standarnya). Pada saat browser mengirimkan header permintaan, sinyal http *request* dikirimkan ke *server*. Header tersebut diterima squid dan dibaca. Dari hasil pembacaan, squid akan memparsing URL yang dibutuhkan, lali URL ini dicocokkan dengan *database cache* yang ada.

Database ini berupa kumpulan metadata (semacam *header*) dari yang sudah ada didalam hardisk. Jika ada, object akan dikirimkan ke klien dan tercatat dalam *logging* bahwa *client* telah mendapatkan *object* yang diminta. Dalam log kejadian tersebut akan dicatat sebagai TCP_HIT. Sebaliknya, jika *object* yang diminta ternyata tidak ada, squid akan mencarinya dari peer atau langsung ke *server* tujuan. Setelah mendapatkan *object*nya, squid akan menyimpan *object* tersebut ke dalam hardisk. Selama dalam proses download *object* ini dinamakan "*object in transit*" yang sementara akan menghuni ruang memori. Dalam masa download tadi, *object* mulai dikirimkan ke klien dan setelah selesai, kejadian ini tercatat dalam log sebagai TCP_MISS.

Hubungan antar *cache* atau nantinya disebut peer itu sendiri ada dua jenis, yaitu *parent* dan *sibling*. *Sibling* kedudukannya saling sejajar dengan *sibling* lainnya, sedangkan *parent* adalah berada diatas *sibling*, dua jenis *peer* ini yang selanjutnya akan bergandengan membentuk jaringan hirarki *cache*.
ICP sebagai protokol *cache* berperan dalam menanyakan ketersediaan *object* dalam *cache*. Dalam sebuah jaringan sebuah *cache* yang mempunyai *sibling*, akan mencoba mencari yang dibutuhkan ke *peer sibling* lainnya, bukan kepada *parent*, *cache* akan mengirimkan sinyal icp kepada *sibling* dan *sibling* membalasnya dengan informasi ketersediaan ada atau tidak. Bila ada, *cache* akan mencatatkan ICP_HIT dalam *log*nya. Setelah kepastian *object* bisa diambil dari *sibling*, lalu *cache* akan mengirimkan sinyal http ke *sibling* untuk mengambil *object* yang dimaksud. Dan setelah mendapatkannya, *cache* akan mencatat *log* SIBLING_HIT.
Jika ternyata *sibling* tidak menyediakan *object* yang dicari, *cache* akan memintanya kepada *parent*. Sebagai *parent*, ia wajib mencarikan *object* yang diminta tersebut walaupun ia sendiri tidak memilikinya (TCP_MISS). Setelah *object* didapatkan dari *server origin*, *object* akan dikirimkan ke *cache child* tadi, setelah mendapatkannya *cache child* akan mencatatnya sebagai PARENT_HIT.

17. *Web Server*

Web server adalah merupakan software yang memberikan layanan data yang berfungsi menerima permintaan HTTP atau HTTPS dari *client* yang

dikenal dengan *browser web* dan mengirimkan kembali hasilnya dalam bentuk halaman - halaman web yang umumnya berbentuk dokumen HTML. *Web server* menunggu permintaan dari *client* yang menggunakan browser seperti Netscape Navigator, Internet Explorer, Modzilla, dan program browser lainnya yang disebut sebagai *User Agent*. Jika ada permintaan dari browser, maka web server akan memproses permintaan itu kemudian memberikan hasil prosesnya berupa data yang diinginkan kembali ke browser. Data ini mempunyai format yang standar, disebut dengan format SGML (*standar general markup language*). Data yang berupa format ini kemudian akan ditampilkan oleh browser sesuai dengan kemampuan browser tersebut. Contohnya, bila data yang dikirim berupa gambar, browser yang hanya mampu menampilkan teks (misalnya lynx) tidak akan mampu menampilkan gambar tersebut, dan jika ada akan menampilkan alternatifnya saja. *Web server*, untuk berkomunikasi dengan *client*-nya (*web browser*) mempunyai protokol sendiri, yaitu HTTP (*hypertext transfer protocol*). Dengan protokol ini, komunikasi antar *web server* dengan *client*-nya dapat saling dimengerti dan lebih mudah.

Seperti telah dijelaskan diatas, format data pada *world wide web* adalah SGML. Tapi para pengguna internet saat ini lebih banyak menggunakan format HTML (*hypertext markup language*) karena penggunaannya lebih sederhana dan mudah dipelajari. Kata *HyperText* mempunyai arti bahwa seorang pengguna internet dengan web browsernya dapat membuka dan membaca dokumen-dokumen yang ada dalam komputernya atau bahkan jauh tempatnya sekalipun. Hal ini memberikan cita rasa dari suatu proses yang tridimensional, artinya pengguna internet dapat membaca dari satu dokumen ke dokumen yang lain hanya dengan mengklik beberapa bagian dari halaman-halaman dokumen (web) itu.

Proses yang dimulai dari permintaan *webclient* (browser), diterima *web server*, diproses, dan dikembalikan hasil prosesnya oleh *web server* ke *web client* lagi dilakukan secara transparan. Setiap orang dapat dengan mudah mengetahui apa yang terjadi pada tiap- tiap proses. Secara garis besarnya web server hanya memproses semua masukan yang diperolehnya dari web clientnya.

Cara kerja web server :

a. Pada saat browser meminta data *web page* ke *server* maka instruksi permintaan data oleh browser tersebut di kemas di dalam TCP yang merupakan protokol *transport* dan dikirim ke alamat yang dalam hal ini merupakan protocol berikutnya yaitu *Hyper Text Transfer Protocol* (HTTP). HTTP ini merupakan protocol yang digunakan dalam *World Wide Web* (WWW) antar komputer yang terhubung dalam jaringan di dunia ini.
b. Data yang di *passing* dari browser ke Web server disebut sebagai HTTP request yang meminta *web page* dan kemudian *web server* akan mencari data HTML yang ada dan di kemas dalam TCP protocol dan di kirim kembali ke browser. Data yang dikirim dari server ke browser disebut sebagai HTTP *response*. Jika data yang diminta oleh browser tidak ditemukan oleh di *web server* maka akan menimbulkan error di *web page* yaitu Error : 404 Page Not Found.

5.1 Jenis-Jenis *Web Server*

5.1.1 Apache

Apache merupakan Aplikasi *Web Server* yang paling banyak digunakan saat ini, selain dapat berjalan di banyak sistem operasi seperti Windows, Linux dan lainya, Apache juga bersifat *Open Source* alias Gratis. Secara *default* Apache menggunakan Script PHP dan menggunakan MySQL sebagai Data Base nya dan kesemuanya itu dapat di gunakan secara gratis pula.

Apache di Ms Windows

Ada beberapa Aplikasi Apache yang dapat berjalan di Windows yaitu :

- Apache2 triad
- XAMPP
- WAMPP
- dll

Apache di Linux

Di Linux pun ada beberapa Aplikasi Apache,yaitu :

- LAMPP
- XAMPP

- Atau kita bisa juga menginstall satu persatu seperti apache(http), PHP dan mysql melalui Source code yang dapat di download di internet secara gratis.

Berikut ini merupakan kelebihan Web Server Apache :

a. Apache termasuk dalam kategori *freeware*.
b. Apache mudah sekali proses instalasinya jika dibanding *web server* lainnya seperti NCSA, IIS, dan lain-lain.
c. Mampu beroperasi pada berbagai *platform* sistem operasi.
d. Mudah mengatur konfigurasinya. Apache mempunyai hanya empat file konfigurasi.
e. Mudah dalam menambahkan peripheral lainnya ke dalam platform *web server*nya.

Berikut ini merupakan kekurangan *Web Server* Apache :

a. *web server* Apache tidak memiliki kemampuan mengatur load seperti IIS, sehingga akan terus mem-fork proses baru hingga nilai MaxClients tercapai atau hingga batas yang diizinkan oleh OS. Ini tentunya menguntungkan penyerang karena habisnya RAM akan lebih cepat tercapai.
b. Apache tidak memproses karakter kutip dalam string *Referrer* dan *User-Agent* yang dikirimkan oleh *client*. Ini berarti *client* dapat memformulasi inputnya secara hati hati untuk merusak format baris log akses.
c. Terganggunya proses upload data, yang bisa menyebabkan *software* salah dalam menerjemahkan ukuran data yang masuk.

5.1.2 Apache Tomcat

Apache tomcat adalah *web server* berbasis *open source* yang mendukung untuk penggunaan JSP (Java Server Pages). Secara default, server tomcat ini belum memiliki package admin, sehingga tidak ada akses untuk admin. Yang ada hanyalah tomcat manager untuk memdeploy aplikasi web melalui file .war dari java web.

Ketika kita berusaha masuk ke URL admin ada keterangan *Tomcat's administration web application is no longer installed by default*. Download and install the "admin" package to use it. Hal itu dikarenakan tidak ada package admin dalam apache tomcat tersebut. Akses administration server tomcat ini bisa untuk setting JNDI. *Java Naming and Directory Interface*

(JNDI) adalah sebuah API yang mendeskripsikan library Java yang standar untuk mengakses layanan *naming* dan *directory* seperti *Domain Name Syatem* (DNS), dan Lightweight Directory Access Protocol (LDAP).

Apache Tomcat merupakan implementasi software open source dari Java Servlet dan JavaServer Pages teknologi. Java Servlet dan JavaServer Pages spesifikasi yang dikembangkan di bawah Java Community Process.

Berikut ini merupakan kelebihan Apache Tomcat :

a. Performanya yang tinggi, stabil, memiliki banyak fitur
b. Mudah dikonfigurasi
c. Menggunakan hanya sedikit sumber daya pada *server*.
d. Tidak bergantung kepada *thread* untuk melayani *client*.

Berikut ini merupakan kekurangan Apache Tomcat :

a. Belum mendukung IPV6
b. *Update/patch* versi terbarunya lama keluar
c. *Fast*-CGInya tidak berfungsi maksimal
d. Pemakainya tidak sebanyak Apache atau IIS(Komunitasnya tidak sebanyak Apache atau IIS)

5.1.3 *Web Server* IIS

IIS atau *Internet Information Services* atau *Internet Information Server* adalah sebuah HTTP *web server* yang digunakan dalam sistem operasi *server* Windows, mulai dari Windows NT 4.0 Server, Windows 2000 Server atau Windows Server 2003. Layanan ini merupakan layanan terintegrasi dalam Windows 2000 Server, Windows Server 2003 atau sebagai add-on dalam Windows NT 4.0. Layanan ini berfungsi sebagai pendukung protokol TCP/IP yang berjalan dalam lapisan aplikasi (*application layer*). IIS juga menjadi pondasi dari *platform* Internet dan Intranet Microsoft, yang mencakup Microsoft Site Server, Microsoft Commercial Internet System dan produk-produk Microsoft BackOffice lainnya.

IIS telah berevolusi semenjak diperkenalkan pertama kali pada Windows NT 3.51 (meski kurang banyak digunakan) hingga IIS versi 6.0 yang terdapat dalam Windows Server 2003. Versi 5.0 diintegrasikan dalam Windows 2000, sedangkan Windows XP Professional memiliki IIS versi 5.1. Windows NT 4.0 memiliki versi 4.01 yang termasuk ke dalam add-on Windows NT Option

Pack. Dalam Windows NT 4.0 Workstation atau Windows 95/98, IIS juga dapat diinstalasikan sebagai Microsoft Personal Web Server (PWS).

Komponen pendukung IIS antara lain :

a. Protocol jaringan TCP/IP
b. Domain Name System(DNS)
c. Direkomendasikan untuk menggunakan format NTFS demi keamanan
d. *Software* untuk membuat situs web, salah satunya Microsoft FrontPage

Fitur-FITUR IIS adalah seperti berikut ini :

a. IIS dapat digunakan sebagai platform dimana aplikasi web berjalan. Hal itu dapat dilakukan menggunakan ASP, ASP.NET, ISAPI,CGI, Microsoft .Net Framework, VBScript, Jscript dan PHP.
b. IIS mendukung protocol HTTP, FTP, SMTP, NNTP dan SSL.
c. IIS mengizinkan aplikasi web untuk dijalankan sebagai proses yang terisolasi dalam ruangan memori terpisah untuk mencegah satu aplikasi membuat crash aplikasi lain.
d. IIS dapat diatur dengan Microsoft Management Console atau menggunakan skrip Windows Scripting Host
e. IIS mendukung bandwidth throttling yang dapat mencegah sebuah situs web memonopoli *bandwidth* yang tersedia.

Berikut ini merupakan kelebihan IIS :

a. Lebih kompatibel dengan Windows karena memang IIS adalah keluaran Microsoft
b. Untuk *platform* .NET, user hanya bisa menggunakan IIS
c. IIS memiliki fitur URL Filtering untuk mem-filter website yang tidak diinginkan

Berikut ini merupakan kekurangan IIS :

a. Tidak gratis (*web server* berbayar)
b. Port 80 (Port untuk layanan web) sangat mudah diserang oleh *cracker*
c. Keamanan file log juga sangat mudah ditembus sehingga sistem password pun akan mudah didapatkan.

5.1.4 *Web server* LIGHTTPD

Web Server yang alamat websitenya di http://lighttpd.net/ ini ditulis dalam bahasa pemrograman C. Dapat dijalankan pada sistem operasi Linux dan

sistem operasi Unix-like lainnya, serta dapat pula dijalankan pada sistem operasi Windows. Lighttpd bersifat *open source* dan didistribusikan dengan lisensi BSD. Lighttpd mengklaim dirinya sebagai *web server* yang "memakan" sedikit space memori jika dibandingkan dengan *web server* lain. Selain itu lighttpd mempunyai kemampuan untuk mengatur *cpu-load* secara efektif dan beberapa fitur advance seperti FastCGI, SCGI, Auth, Output-Compression, URL-Rewriting dan lain-lain. Secara lengkap, fitur-fitur yang disediakan oleh lighttpd adalah sebagai berikut:

a. Dukungan *load-balancing* FastCGI, SCGI, dan HTTP proxy.
b. Dukungan chroot.
c. *Web server* berbasis select()-/poll()-/epoll()
d. Dukungan untuk skema notifikasi *event* yang lebih efisien seperti *queue* dan *epol*.
e. *Conditional rewrites (mod_rewrite)*.
f. Dukungan SSL dan TLS, via OpenSSL.
g. Otentikasi terhadap sebuah *server* LDAP
h. Statistik RRDtool
i. *Rule-based downloading* dengan kemungkinan penanganan sebuah *script* hanya otentikasi
j. Dukungan *Server Side Includes*
k. Dukungan *modules*
l. Dukungan minimal WebDAV

Berikut ini merupakan kelebihan Lighttpd :

a. *Virtual hosting* yang fleksibel
b. HTTP *compression* menggunakan mod_compress dan mod_deflate terbaru (1.5.x)
c. Berukuran kecil (kurang dari 1 MB)
d. Desain *single-process* hanya dengan beberapa *thread*. Tidak ada proses atau *thread* dimulai per koneksi.

Berikut ini kekurangan Lighttpd :

a. Belum support IPV6
b. *Update/patch* versi terbarunya lama keluar
c. *Fast*-CGInya tidak berfungsi maksimal

d. Pemakainya tidak sebanyak Apache atau IIS (komunitasnya tidak sebanyak Apache atau IIS)

18. *Email Server*

Email atau *electronic-mail* adalah suatu bentuk komunikasi dengan menggunakan perangkat elektronik terutama komputer. *Server* dalam Internet menjalankan sebuah aplikasi yang akan menunggu program untuk mengirimkan data atau perintah ke *server* tersebut. *Server* email menjalankan sebuah aplikasi yang ditujukan untuk proses pengiriman dan penerimaan email. Aplikasi yang berjalan pada *server* ini disebut dengan MTA (*Mail Transfer Agent*) sedangkan aplikasi yang berjalan pada komputer-komputer lain yang dilayani oleh server (*client*) disebut dengan MC (*Mail Client*).

Ada banyak *server* email yang saat ini digunakan dalam jaringan Internet yang menggunakan sistem operasi Linux/Unix. Tiga di antaranya yang terkenal adalah SendMail, QMail, dan Postfix. Ketiga *server* email ini memiliki keunggulan dan kelemahan masing-masing dalam melayani dan mengirimkan email. Sangat menarik melihat keunggulan ketiganya ini terutama dalam bidang pengamanan (*security*).

6.1 *Server* SENDMAIL

Server email SendMail adalah sebuah *server* email standar yang satu paket (*built-in*) dengan sistem operasi Linux/Unix. SendMail ini merupakan *server* email yang paling dahulu muncul di antara ketiga server email. Ada banyak versi SendMail yang sudah digunakan. Versi pertama adalah 8.8.5 dan diluncurkan tanggal 21 Januari 1997 [http://www.sendmail. org].

Server email SendMail dirancang dalam sebuah program besar. Program ini menjalankan semua fungsi SendMail sebagai sebuah *server* email. Program besar ini sangat mudah dalam membagi data oleh bagian-bagian sistem yang berbeda. Namun, jika terjadi kesalahan fatal atau masuknya pengganggu yang merusak sistem, maka sistem akan terganggu sehingga semua fungsi server email juga ikut terganggu.

6.1.1 Pengamanan (*Security*)

SendMail merupakan *server* email yang paling banyak memiliki lubang pengamanan sehingga bisa ditembus oleh kode-kode jahat. Hal ini diperkuat



oleh data dari *Internet Security Systems* yang bekerja sama dengan *Departemen of Homeland Security*. Data terbaru mengatakan bahwa tanggal 3 Maret 2003, ditemukan sebuah lubang pada root SendMail untuk versi yang lebih rendah dari 8.12.8 sehingga memaksa setiap sistem pada jaringan di-*upgrade* ke versi 8.12.8 atau yang lebih tinggi.

Tidak itu saja lubang-lubang pengamanan yang ada pada SendMail. Masih banyak lagi lubang lain yang muncul pada setiap versi SendMail.

6.2 SERVER QMAIL

Server email QMail adalah sebuah *server* email yang harus di-*install* sendiri ke dalam sistem operasi Linux/Unix. QMail ini merupakan *server* email jawaban yang oleh pembuatnya disebut sebagai *server* email teraman karena sampai sekarang belum ditemukan satupun lubang pengamanan yang bisa mengganggu kinerjanya. QMail dibuat oleh Dan J. Bernstein, seorang profesor Departemen Matematika, Statistik, dan Ilmu Komputer, Universitas Illinois, Chicago. QMail sudah banyak digunakan oleh host-host ternama seperti Hotmail dan Yahoo!Mail.

Server email QMail dirancang dalam beberapa program kecil yang terhubung dalam sebuah hierarki kaku. Di sini program dalam hierarki bisa menjalankan program-program lain dalam urutan yang tetap dan membuang program itu dalam urutan jika sudah selesai. Cara ini bisa memberikan penyekatan yang lebih baik jika dibandingkan dengan SendMail, tetapi di sisi lain adanya penciptaan proses dan komunikasi antar-proses membutuhkan biaya tambahan. Biaya tambahan ini bisa ditekan dalam batasan tertentu dengan mempartisi pekerjaan menurut cara tertentu pula.

6.2.1 Pengamanan (*Security*):

QMail diklaim sebagai *server* email yang paling aman karena tidak ditemukannya lubang pengamanan yang bisa diterobos oleh pihak-pihak penyerang hingga kini. Berikut adalah pengamanan yang terdapat pada QMail:

a. Secara optional dapat menambahkan deskripsi *header* X-Qmail-Scanner untuk setiap pesan e-mail yang telah melalui pemeriksaan.

b. Pesan e-mail yang mengandung virus akan dipindahkan kedalam direktori mailder, agar selanjutnya pembacaan pesan tersebut dapat dilakukan oleh pihak yang tepat.
c. Dapat diintegraskan dengan spamAssasins, sehingga dapat melakukan Anti Spamming yang kompherensif.
d. Mem-block alamat e-mail berdasarkan tipe file attachment-nya, atau dapat juga berdasarkan header pesan tertentu di dalam e-mail tersebut, misalnya : file attachment berekstensi *.mp3, *.exe, atau yang mempunyai subyek : ILOVEYOU.

6.3 *Server* P OSTFIX

Server email Postfix adalah sebuah *server* email yang dikembangkan sebagai pengganti SendMail dan bisa dijalankan dengan baik di sistem operasi Unix dan Max O/S X. Postfix ini dibuat oleh Wietse Venema, seorang ahli pengamanan (*security specialist*) saat bekerja sebagai peneliti di IBM. Postfix merupakan versi yang dibagikan secara gratis dari perusahaan email komersial IBM, *Secure Mailer* [http://www.daily_daemonnews.org/view_story.php3]. Pertama kali dibuat tahun 1998 dan menjadi populer karena mudah mendapatkan kode programnya dan dijalankan dalam sistem operasi terbuka. Postfix dipasarkan mulai tanggal 22 Januari 1999. *Server* email Postfix dirancang dalam beberapa program kecil. Proses-proses yang ada bersifat semi-tetap. Proses-proses saling bekerja sama dalam melakukan task dalam sebuah kerja sama sejajar, bukan dalam bentuk hubungan *parent-child*. Di samping itu, Postfix mempunyai pelayanan untuk setiap program kecilnya sehingga tidak perlu mengeluarkan biaya untuk membentuk pelayanan itu.

6.3.1 Pengamanan (*Security*):

Server email Postfix adalah sebuah *server* email yang paling muda di antara ketiga *server* email yang dibahas dalam makalah ini. Postfix dianggap sebagai MTA yang jauh lebih aman daripada SendMail dan lebih cepat daripada QMail. Pengamanan yang ada pada server email Postfix ini adalah :

1. Hak-hak istimewa terbatas

   Hampir semua program daemon Postfix bisa dijalankan dengan hak istimewa terbatas yang tetap dalam lingkungan chroot. Hal ini sangat mendukung untuk program-program yang muncul dalam jaringan: server SMTP dan client SMTP. Walaupun tidak ada jaminan kerja sama dengan sistem, fitur ini tetap bisa menambah hambatan bagi pengamanan sistem meskipun sedikit.

2. Penyekatan

   Postfix menggunakan proses-proses terpisah untuk menyekat kejadian satu sama lain. Hal ini terlihat dari tidak adanya jalur langsung dari jaringan ke program penyebaran pengamanan lokal (*security-sensitive local delivery programs*) sehingga pengganggu harus melewati banyak program lain dulu. Di sini ada beberapa bagian Postfix yang bersifat banyak proses (*multi-threaded*). Akan tetapi, semua program yang berhubungan dengan dunia luar tetap bersifat satu proses (*single-threaded*). Proses-proses terpisah ini memberikan penyekatan yang lebih baik daripada banyaknya proses dalam satu ruang alamat yang dibagi.

3. Lingkungan yang terkontrol

   Tidak ada program pengantaran email yang berjalan di bawah kendali proses pengguna. Sebagai gantinya, program-program Postfix berjalan dalam sebuah daemon master yang tetap yang berjalan dalam sebuah lingkungan yang terkendali, tanpa adanya hubungan *parent-child* dengan proses pengguna. Keuntungannya, pendekatan ini bisa menurunkan penggunaan atribut-atribut proses UNIX/Linux yang lain, sinyal, pembukaan file-file, variabel lingkungan sehingga sistem UNIX melewati kemungkinan penurunan *parent* yang berbahaya ke *child*.

4. Set-uid

   Semua program Postfix diatur sebagai set-uid. Konsep uid ini muncul karena adanya kesalahan pada UNIX. Awalnya, set-uid dan set-gid diharapkan memberikan pengaruh yang bagus pada UNIX, tetapi yang didapatkan malah sebaliknya. Setiap kali program baru dimasukkan ke UNIX, set-uid selalu menimbulkan masalah pengamanan seperti program *shared libraries*, sistem file /proc, dan dukungan beragam bahasa. Hal ini dikarenakan set-uid tidak mampu memperkenalkan fitur-fitur tersebut. Awalnya, direktori antrian

maildrop bersifat bisa ditulis sehingga proses-proses lokal bisa mengirimkan mail tanpa bantuan dari perintah set-uid atau set-gid atau dari proses mail daemon. Direktori maildrop tidak digunakan untuk mail yang datang dari jaringan, dan file-file antriannya tidak bisa dibaca untuk pengguna yang tidak memiliki hak.

Direktori yang bersifat bisa ditulis ini ternyata memberikan peluang gangguan di mana pengguna lokal akan mampu melakukan link dengan file-file maildrop orang lain sehingga file-file itu tidak akan bisa diakses atau disebarkan. Pengguna lokal bisa mengisi direktori maildrop dengan sampah (mail-mail tidak berguna) dan mungkin saja berusaha membuat agar sistem mengalami *crash*. Di samping itu mereka juga bisa memindahkan file-file orang lain ke dalam direktori maildrop dan menyebarkannya sebagai mail. Namun demikian, file-file antrian Postfix memiliki format tertentu, di mana kurang dari satu di antara 10^{12} file-file non-Postfix akan dikenali sebagai file antrian Postfix yang sah.

Karena kemungkinan timbulnya bahaya akibat direktori maildrop yang bisa ditulis ini, maka Postfix menggunakan program bantuan set-gid postdrop untuk perizinan mail.

5. Kepercayaan

   Program-program Postfix tidak akan mudah mempercayai isi file-file antrian atau pesan IPC internal Postfix. File-file antrian ini tidak memiliki catatan pada disk untuk disebarkan ke tujuan-tujuan tertentu seperti ke file atau perintah program. Sebagai gantinya, program-program seperti agen *delivery* lokal akan berusaha menghasilkan keputusan strategis berdasarkan informasi tangan pertama.

   Tentu saja, program-program Postfix tidak akan mempercayai data yang berasal dari jaringan. Secara teknis, Postfix melakukan penyaringan data yang dikirim berikut pengirimnya sebelum menyebarkannya melalui variabel internet. Ini adalah pelajaran yang harus diperhatikan oleh seseorang dalam hal pengamanan, yaitu: jangan biarkan data apapun dari jaringan yang berada dekat sebuah shell. Penyaringan ini merupakan hal terbaik yang bisa dilakukan untuk mengatasi hal ini.

6. Input yang besar

Memori untuk string dan buffer dialokasikan secara dinamis untuk mencegah terjadinya permasalahan *overrun* pada buffer. Baris yang panjang pada input pesan akan dipecah/dipotong menjadi beberapa urutan terhubung dengan ukuran yang lebih kecil. Pecahan ini kemudian disatukan lagi pada saat pengiriman. Diagnosa pemotongan ini dilakukan dalam satu tempat kemudian dikirim ke antarmuka syslog(3) untuk mencegah terjadinya *overrun* buffer pada sistem operasi yang lebih lama. Namun demikian, tidak ada cara umum yang digunakan untuk memotong data sebelum dilewatkan ke *system call* atau ke rutin *library*. Pada beberapa sistem operasi, perangkat lunak mungkin masih mengalami masalah overrun buffer, karena kerentanan perangkat lunak utamanya.

Sampai saat ini tidak ada cara khusus argumen baris perintah yang panjang. Kernel UNIX menentukan batasan sendiri yang diharapkan cukup memenuhi program-program yang berjalan atau menghadapi pengguna-pengguna jahat.

7. pengamanan-pengamanan yang lain

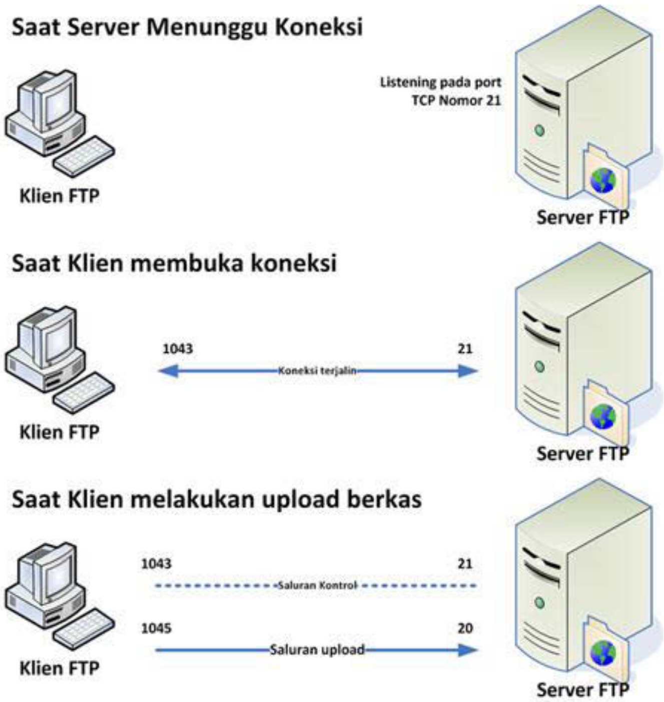
   Jumlah tipe-tipe objek yang menggunakan memori dibatasi untuk mencegah sistem mail menjadi padat karena beratnya beban. Di samping itu, jika terjadi masalah, maka perangkat lunak akan menghentikan pengiriman untuk beberapa lama sebelum terjadinya error yang fatal atau sebelum menjalankan kembali program (*restart*) yang gagal. Tujuannya adalah untuk mencegah kondisi-kondisi yang jika terus dijalankan akan mengakibatkan terjadinya hal yang lebih buruk lagi.

19. *File Transfer Protocol* (FTP)

FTP (singkatan dari *File Transfer Protocol*) adalah sebuah protokol Internet yang berjalan di dalam lapisan aplikasi yang merupakan standar untuk pentransferan berkas (*file*) komputer antar mesin-mesin dalam sebuah *internetwork*. FTP merupakan salah satu protokol Internet yang paling awal dikembangkan, dan masih digunakan hingga saat ini untuk melakukan pengunduhan (*download*) dan penggugahan (upload) berkas-berkas komputer antara *client* FTP dan *server* FTP. Sebuah *client* FTP merupakan aplikasi yang dapat mengeluarkan perintah-perintah FTP ke sebuah *server*

FTP, sementara *server* FTP adalah sebuah Windows Service atau daemon yang berjalan di atas sebuah komputer yang merespons perintah-perintah dari sebuah *client* FTP. Perintah-perintah FTP dapat digunakan untuk mengubah direktori, mengubah modus transfer antara biner dan ASCII, menggugah berkas komputer ke server FTP, serta mengunduh berkas dari server FTP.

Sebuah *server* FTP diakses dengan menggunakan *Universal Resource Identifier* (URI) dengan menggunakan format ftp://namaserver. Klien FTP dapat menghubungi server FTP dengan membuka URI tersebut.



Gambar 4 Cara kerja FTP *server*

FTP menggunakan protokol *Transmission Control Protocol* (TCP) untuk komunikasi data antara *client* dan *server*, sehingga di antara kedua komponen tersebut akan dibuatlah sebuah sesi komunikasi sebelum transfer data dimulai. Sebelum membuat koneksi, port TCP nomor 21 di sisi *server* akan "mendengarkan" percobaan koneksi dari sebuah *client* FTP dan kemudian akan digunakan sebagai port pengatur (*control port*) untuk (1) membuat sebuah koneksi antara *client* dan *server*, (2) untuk mengizinkan klien untuk mengirimkan sebuah perintah FTP kepada server dan juga (3) mengembalikan respons *server* ke perintah tersebut. Sekali koneksi kontrol telah dibuat, maka *server* akan mulai membuka port TCP nomor 20 untuk

membentuk sebuah koneksi baru dengan klien untuk mentransfer data aktual yang sedang dipertukarkan saat melakukan pengunduhan dan penggugahan.

Ketika kita menggunakan FTP (sebagai user) terdapat dua jenis atau cara yang dapat dilakukan:

a. *Anonymous*, Sistem FTP *anonymous* sudah sejak lama diciptakan dengan tujuan agar setiap orang yang terkoneksikan ke dalam dunia internet dapat saling berbagi file dengan orang lain yang belum memiliki account dalam server. Dengan sistem ini setiap orang dapat menggunakan sebuah account yang umum (*public account*) berupa *anonymous*. Mempunyai hal keterbatasan hak akses. Keterbatasan yang dimiliki ketika pengguna menggunakan jenis FTP *anonymous* biasanya meliputi keterbatasan dalam proses akses direktori dan file yang tersedia dalam *server* yang dituju. Selain itu, pengguna yang menggunakan sistem ini tidak dapat melakukan *uploading* data terhadap *server* yang dituju. Namun sebaliknya ia hanya memiliki kemampuan dalam *downloading*, baca file tertentu dan pindah direktori yang diizinkan oleh pemilik *server*.

b. *User legal* (*authenticated user*), adalah sebuah cara lain yang digunakan oleh pengguna internet dalam mengakses sebuah *server* dengan menggunakan FTP. Untuk dapat mengkases remote host, cara *user legal* (*authenticated user*) menuntut kita untuk memiliki sebuah *account* khusus yang dimiliki secara pribadi. Untuk dapat memiliki *account* khusus ini, seorang pengguna harus mendaftarkan diri terlebih dahulu kepada pemilik *remote host* tersebut. Terdapat banyak *server* yang memberikan fasilitas *account* FTP secara gratis, selain *server* lain yang mengharuskan pengguna untuk membeli sebuah *account* yang tentunya dengan fasilitas yang lebih banyak dibandingkan dengan sebuah *account* yang dimiliki secara gratis. Dengan menggunakan *account* ini, seorang pengguna memiliki hak akses yang jauh berbeda dengan seorang pengguna jenis *anonymous*. Selain kemampuan yang dimiliki oleh pengguna *anonymous*, seperti download dan berpindah dari satu direktori ke direktori lain serta kemampuan baca file tertentu, *uploading*, membuat sebuah direktori, menghapus file dan direktori. Hak yang dimiliki adalah hak seorang pemilik bukan seorang pengunjung biasa.

Contoh aplikasi FTP server :

a. Proftpd
b. Vsftpd
c. Wuftpd
d. IIS (didalamnya terdapat FTP Server)

Contoh aplikasi FTP client :

a. CuteFTP, Wget
b. WsFTP
c. GetRight
d. AbsoluteFTP
e. SmartFTP
f. Filezilla(Mendukung SFTP)

c. Rangkuman

Di sisi server, layanan sistem operasi jaringan diantaranya:

1. DNS Server atau Domain Name System adalah distribute database system yang digunakan untuk pencarian nama komputer di jaringan yang mengunakan TCP/IP (Transmission Control Protocol/Internet Protocol). DNS server berfungsi untuk menterjemahkan (mentranslasi) IP address sebuah server menjadi nama domain. DNS digunakan pada aplikasi yang terhubung ke Internet seperti web browser atau e-mail, dimana DNS membantu memetakan host name sebuah komputer ke IP address. DNS dapat disamakan fungsinya dengan buku telepon. Dimana setiap komputer di jaringan Internet memiliki host name (nama komputer) dan Internet Protocol (IP) address. Secara umum, setiap client yang akan mengkoneksikan komputer yang satu ke komputer yang lain, akan menggunakan host name. Lalu komputer anda akan menghubungi DNS server untuk mencek host name yang anda minta tersebut berapa IP address-nya. IP address ini yang digunakan untuk mengkoneksikan komputer anda dengan komputer lainnya. Sebagai contoh,http://ersprayogapangestu221.blogspot.com/ mempunyai IP address 223.255.219.30. Ketika seseorang mengakses blog tersebut yang berarti merequest situs http://aantomatika.blogspot.com, DNS request tersebut akan diarahkan oleh DNS server ke server yang memiliki IP address 223.255.219.30.

2. DHCP Server DHCP atau Dynamic Host Configuration Protocol adalah sebuah layanan yang secara otomatis memberikan nomor IP kepada komputer yang memintanya. Komputer yang memberikan nomor IP inilah yang disebut sebagai DHCP server, sedangkan komputer yang melakukan request disebut DHCP Client.
3. Proxy server adalah server yang berfungsi melayani permintaan dokumen World Wide Web dari pengguna internet, dapat menyimpan sementara (cache) file html server lain untuk mempercepat akses internet. Dapat dikatakan proxy server bekerja dengan menjembatani komputer (program internet seperti browser, download manager dan lainnya) ke internet. Demikian sedikit bahasan tentang layanan (services) pada sistem operasi jaringan, presentasinya dapat diunduh di sini.
4. Web Server adalah layanan server yang berfungsi menerima permintaan HTTP atau HTTPS dari klien yang dikenal dengan web browser dan mengirimkan kembali hasilnya dalam bentuk halaman-halaman web yang umumnya berbentuk dokumen HTML. Salah satu server web yang terkenal di linux adalah Apache. Apache merupakan server web antar platform yang dapat berjalan di beberapa platform seperti Linux dan Windows. Web Server juga merupakan sebuah komputer yang menyediakan layanan untuk internet. Server disebut juga dengan host. Agar anda dapat memasukkan web yang anda rancang ke dalam internet, maka anda harus memiliki ruangan terlebih dahulu dalam internet (hosting), ruangan inilah yang disediakan oleh server.
5. Mail Server Mail server yaitu layanan atau perangkat lunak program yang mendistribusikan file atau informasi sebagai respons atas permintaan yang dikirim via email, juga digunakan pada bitnet untuk menyediakan layanan serupa http://FTP.
6. FTP Server FTP (File Transfer Protocol) server adalah layanan sistem operasi yang berfungsi untuk memberikan layanan tukar menukar file dimana server tersebut selalu siap memberikan layanan FTP apabila mendapat permintaan (request) dari FTP client. FTP client adalah computer yang merequest koneksi ke FTP server untuk tujuan tukar menukar file berupa download, upload, rename file, deleting file, dll sesuai dengan permission yang diberikan oleh FTP server. Tujuan dari FTP server adalah



sebagai berikut : sharing data menyediakan indirect atau implicit remote computer menyediakan tempat penyimpanan bagi user menyediakan transfer data yang reliable dan efisien

d. Tugas

n. Sebutkan macam-macam layanan jaringan dan jelaskan secara singkat!
o. Apa fungsi dari firewall?
p. Apa fungsi dari DNS?
q. Apa fungsi dari DHCP?
r. Apa yang dimaksud dengan IP masquerade?
s. Bagaimana proxy server bekerja?
t. Kenapa caching dilakukan oleh proxy server?
u. Bagaimana cara kerja web server?
v. Apa saja yang bisa dilakukan oleh e-mail server?
w. Apa saja yang difasilitasi oleh FTP?

e. Test Formatif

1. Apa saja yang bisa dilakukan oleh DNS server?
   a) Memetakan alamat IP menjadi nama domain.
   b) Memetakan alamat IP menjadi mac address.
   c) Memetakan nama domain menjadi alamat IP.
   d) Jawaban B dan C benar.
   e) Salah semua.
2. Apa yang diberikan oleh DHCP server?
   a) Alamat IP.
   b) Nama domain.
   c) Mac address.
   d) Nomor port.
   e) Akses internet.



3. Apa yang digunakan sebagai firewall di sistem Linux?
    a) Firebug
    b) IPTables
    c) IPProxy
    d) Firewell
    e) Wallfire
4. Proxy server melakukan caching, yang dimaksud dengan caching adalah ...
    a) Menyimpan informasi
    b) Mengakses informasi untuk dibagikan kepada peminta informasi lainya
    c) Menyimpan data untuk dibagikan kepada peminta tanpa perlu meminta ulang data di jaringan
    d) Mengirim data melalui internet tanpa perlu meminta pengguna untuk mengijinkan pengiriman data
    e) Menutup jaringan informasi
5. Apa protocol yang digunakan oleh web server?
    a) DHCP
    b) FTP
    c) SUCP
    d) HTTP
    e) HTML
6. Layanan web server bisa diakses melalui ... oleh pengguna.
    a) Mobile
    b) PC
    c) User Agent
    d) Penjelajah
    e) HTTP
7. Berikut yang bukan termasuk HTTP server adalah ...
    a) Apache Tomcat
    b) Glassfish
    c) Apache
    d) IIS
    e) NetOpenHTTP

8. Apa saja protocol yang digunakan oleh Email server?
    a) SMTP
    b) POP
    c) FTP
    d) Jawaban A dan B.
    e) Jawaban A dan C.
9. Bagaimana bentuk URL untuk FTP?
    a) ftp/localhost
    b) ftp//localhost
    c) ftp:/localhost
    d) ftp://localhost
    e) ftp::/localhost
10. Berikut yang bukan termasuk FTP server adalah ...
    a. Proftpd
    b. FTPShrine
    c. IIS
    d. Wuftpd
    e. Salah semua

f. Lembar Jawaban Tes Formatif

g. Lembar Kerja Siswa

12. Kegiatan Belajar 12 : Menyajikan Berbagai Layanan Jaringan
Tujuan Pembelajaran

Setelah mengikuti kegiatan belajar 12 ini, siswa diharapkan dapat:

x. Menyajikan berbagai layanan jaringan dalam bentuk laporan, makalah atau presentasi.

Uraian Materi

Siswa diharapkan bisa menyajikan bebagai layanan jaringan dalam bentuk laporan, makalah atau presentasi yang dilakukan dalam berkelompok. Kelompok tediri dari minimal 3 sampai maksimal 5 orang untuk menjaga efisiensi.

Laporan menjadi tugas bagi siswa untuk melakukan analisa mendalam terhadap jenis-jenis layanan jaringan. Sedangkan makalah memuat detail rinci tentang salah satu jenis layanan jaringan. Hasil makalah tersebut diolah untuk dijadikan presentasi sehingga menjadi padat, efektif untuk disampaikan di depan publik dalam waktu yang singkat.

Rangkuman

Tugas

1. Buatlah kelompok yang terdiri dari 3 – 5 orang.
2. Buatlah sebuah laporan tentang salah satu merk dari penyedia semua layanan jaringan, misalnya Apache untuk web server, lalu Mercury untuk E-mail server, dst. Sebutkan keunggulan masing-masing dan kelemahan masing-masing.
3. Buatlah makalah tentang salah satu layanan jaringan, dan jelaskan secara rinci tentang sejarah singkat, cara kerja, kegunaan, dan implementasi di dunia nyata dan contoh kasus penggunaan apabila digunakan di lingkup sekolah (Tidak boleh sama dengan kelompok lain).
4. Sajikan makalah yang sudah dibuat dalam bentuk presentasi.

Tes Formatif

Lembar Jawaban Tes Formatif

Lembar Kerja Siswa

13. Kegiatan Belajar 1 : Memahami Manajemen Backup dan Recovery pada Linux

Tujuan Pembelajaran

Setelah mengikuti kegiatan belajar 1 ini, siswa diharapkan dapat :

1.Memahami manajemen backup dan recovery pada Linux.

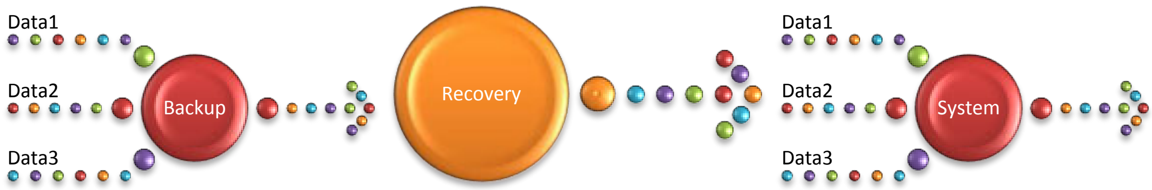
a. Uraian Materi

1.Manajemen Backup dan Recovery

20. Backup dan Recovery

Sebuah sistem yang besar dan krusial harus selalu diatur setiap waktunya. Bukan hanya diatur untuk selalu mendapatkan data yang baru, tapi juga diatur supaya data yang lama tetap terjaga dengan aman.

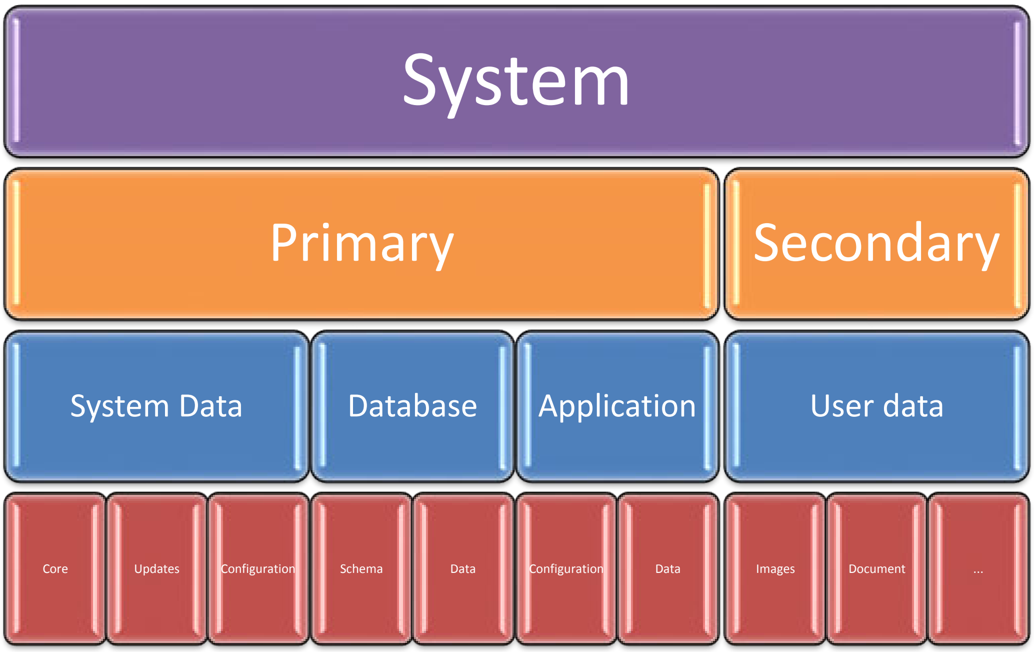
Salah satu caranya adalah dengan melakukan *backup* dan *recovery*. Backup dan recovery merupakan kata yang saling berkaitan. Backup berarti kita membuat cadangan dari data yang sudah ada untuk disimpan dalam bentuk yg lain atau sama. Recovery, adalah merubah bentuk dari cadangan untuk dikembalikan menjadi data yang semula sudah ada.



Banyak data yang sudah ada di backup, lalu apabila dibutuhkan lagi akan dilakukan recovery sehingga data-data tersebut kembali seperti semula kepada sistem.

Data-data tersebut biasanya diklasifikasikan. Jenis klasifikasi ini tidak baku, namun yang paling baik adalah klasifikasi berdasarkan konten.

Jadi kita bisa pisahkan data-data di sistem menjadi klasifikasi seperti berikut.



Kita mengklasifikasikan data sistem menjadi dua kelompok, primer dan sekunder. Data primer ini sangat penting dan sistem bergantung kepadanya. Semua inti dari sistem, baik saat masih baru atau beserta update, serta konfigurasi tambahan yang diperlukan. Data database, mulai dari skema dari database dan isinya. Lalu disertai dengan data aplikasi beserta konfigurasi dan datanya. Menyusul data sekunder yang berisi data pribadi pengguna yang tidak berdampak langsung kepada sistem, tapi berdampak kepada pengguna.

Pembagian file backup pun juga dibagi berdasarkan klasifikasi ini, jadi kita bisa tidak langsung melakukan backup satu sistem penuh. Kita pilih untuk melakukan backup data system, lalu biarkan satu file backup itu hanya berisi data sistem, jangan dicampuri dengan backup lainya.

Hal ini ditujukan supaya hendak dilakukan recovery, tidak semua data berubah kembali menjadi seperti semula, mungkin hanya ada kesalahan pada database, dan kita ingin mengembalikanya. Dengan pembagian seperti di atas, maka kita bisa hanya mengembalikan data database dan data lainya tetap baru dan berjalan lancar.

Selain itu, backup dalam skala yang kecil juga berjalan lebih cepat, dan juga bisa dipecah-pecah untuk diberikan kepada bagian-bagian yang bertanggung jawab untuk data tersebut.

Meskpun begitu, melakukan backup dalam skala besar keseluruhan sistem juga terkadang dianjurkan. Ini dapat dilakukan apabila ada media penyimpanan yang cukup besar, dan waktu yang tidak terbatas untuk melakukan backup. Apabila waktu dan media penyimpanan yang ada sangat terbatas, dianjurkan untuk memecah-mecah konsentrasi backup.

Kebanyakan sistem sudah mempunyai media dan waktu yang cukup, jadi memecah-mecah konsentrasi bukan lagi pilihan. Melakukan backup satu disk penuh adalah satu-satunya cara yang paling efisien apabila disertai dengan infrastruktur yang memadai.

Backup biasanya dilakukan secara berkala, dengan waktu yang teratur. Semisal setiap 2 bulan sekali, atau apabila sistem benar-benar penting, kompak, dan berubah dengan cepat, maka backup per hari bukanlah hal yang tidak mungkin.

21. **Alasan Kenapa Backup Penting**

Hilangnya file-file yang penting sangat mempengaruhi jalanya suatu kegiatan yang bergantung terhadap file tersebut. Dengan backup, ketakutan akan kehilangan file tersebut sedikit berkurang. Ada beberapa hal yang bisa membuat file hilang.

a. Kegagalan Hardware

b. Salah Hapus
c. Pencurian
d. Virus

Apabila terjadi kegagalan hardware, seperti disk yang rusak, jatuh, terkena air. Ini sudah pasti sangat susah untuk dikembalikan. Sebab lainya adalah salah hapus, meskipun salah hapus, beberapa aplikasi ada yg dikhususkan untuk mengembalikan file yang terhapus. Pencurian, secara fisik, akan mengakibatkan hardware dan software ikut hilang. Virus juga bisa merusak file, menghancurkan file tersebut sampai titik di mana file tersebut tidak bisa dikembalikan lagi.

Oleh karena itu, melakukan backup sangatlah penting, dan usahakan tempat penyimpanan backup tersimpan aman. Server cloud atau hardisk eksternal merupakan solusi yang cukup baik.

22. **Backup dan Recovery di Sistem Linux Debian**

Sistem Linux Debian juga diberi kemampuan untuk melakukan backup dan recovery untuk sistem. Dengan aplikasi **DD** (Dataset Definition), bisa dilakukan sebuah backup penuh terhadap satu disk yang langsung bisa ditaruh disk lainya atau diteruskan sebagai output yang nantinya akan diproses menjadi sebuah file. Sebelum memulai backup, pilih disk yang akan di backup. Untuk melihat daftar disk bisa dicari di direktori **/dev**.

Gunakan,

```
ls –l /dev | more
```

Untuk melihat isi dari direktori **/dev** secara rinci, dan juga membatasi outputnya supaya bisa dibaca. Kegunaan dari operator **I** (pipa) adalah untuk mengarahkan output ke perintah di sebelah kanan dari operator. Dalam kasus ini **more**, digunakan untuk melihat file sedikit demi sedikit.

```
crw------- 1 root root    254,   0 Dec 16 09:12 rtc0
lrwxrwxrwx 1 root root           3 Dec 16 09:12 scd0 -> sr0
brw-rw---- 1 root disk      8,   0 Dec 16 09:12 sda
brw-rw---- 1 root disk      8,   1 Dec 16 09:12 sda1
brw-rw---- 1 root disk      8,   2 Dec 16 09:12 sda2
brw-rw---- 1 root disk      8,   5 Dec 16 09:12 sda5
brw-rw---- 1 root disk      8,   6 Dec 16 09:12 sda6
brw-rw---- 1 root disk      8,  16 Dec 16 09:12 sdb
crw-rw---- 1 root cdrom    21,   0 Dec 16 09:12 sg0
crw------- 1 root root     21,   1 Dec 16 09:12 sg1
crw------- 1 root root     21,   2 Dec 16 09:12 sg2
drwxrwxrwt 2 root root          60 Dec 16 09:12 shm
crw------- 1 root root     10, 231 Dec 16 09:12 snapshot
drwxr-xr-x 3 root root         160 Dec 16 09:12 snd
lrwxrwxrwx 1 root root          24 Dec 16 09:12 sndstat -> /proc/asound/oss/snds
tat
brw-rw---- 1 root cdrom    11,   0 Dec 16 09:12 sr0
lrwxrwxrwx 1 root root          15 Dec 16 09:12 stderr -> /proc/self/fd/2
lrwxrwxrwx 1 root root          15 Dec 16 09:12 stdin -> /proc/self/fd/0
lrwxrwxrwx 1 root root          15 Dec 16 09:12 stdout -> /proc/self/fd/1
crw-rw-rw- 1 root root      5,   0 Dec 16 09:12 tty
crw------- 1 root root      4,   0 Dec 16 09:12 tty0
crw------- 1 root tty       4,   1 Dec 16 09:18 tty1
crw------- 1 root root      4,  10 Dec 16 09:12 tty10
--More--
```

Tekan enter untuk menggeser ke bawah, dan cari bagian yang ada tulisan **disk**. Yang berarti file tersebut adalah representasi dari disk yang ada di sistem kita.

Dalam contoh ini, ada disk dengan nama **sdb** yang berisi data dummy untuk dibackup ke dalam file. Dalam kasus nyata, data yang dibackup seharusnya data penting dan tidak diarahkan ke dalam file, tapi ke media yang lebih aman dan terjamin.

Gunakan perintah dari program **dd**,

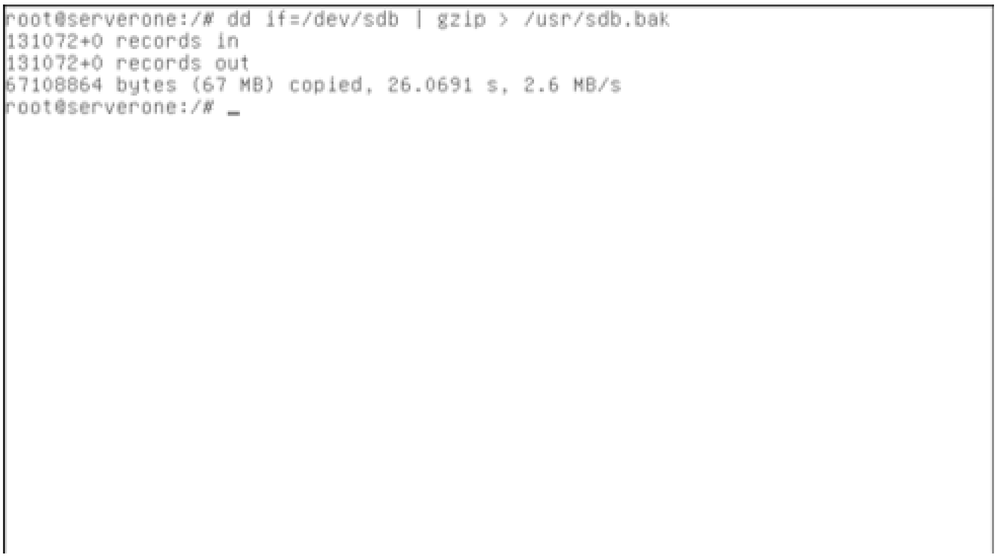
```
dd if=/dev/sdb | gzip > /usr/sdb.bak
```

Kita menggunakan **dd**, dengan input file **/dev/sdb** dan hasilnya di arahkan ke program **gzip** untuk melakukan kompresi data dan disimpan di file **/user/sdb.bak**.

Apabila ingin mencoba untuk melakukan backup langsung ke media, gunakan perintah ini.

```
dd if=/dev/sdb of=/dev/<media_backup>
```

Lalu biarkan proses berjalan, proses akan memakan waktu cukup lama tergantung dari jumlah data yang diproses.



```
root@serverone:/# dd if=/dev/sdb | gzip > /usr/sdb.bak
131072+0 records in
131072+0 records out
67108864 bytes (67 MB) copied, 26.0691 s, 2.6 MB/s
root@serverone:/# _
```

Terlihat bahwa sudah berhasil dilakukan backup pada disk **sdb** ini. Ukuran dari disk **sdb** ini hanya 64MB sehingga proses tidak membutuhkan waktu terlalu lama, yaitu cuma 26.0691 detik.

Sekarang kita sudah berhasil membuat backup. Sekarang kita membutuhkan cara untuk mengembalikan hasil backup kita (recovery) ke disk semula.

Gunakan perintah ini untuk melakukan recovery dari data yang sudah di backup menggunakan kompresi gzip.

```
gzip –dc /usr/sdb.bak | dd of=/dev/sdb
```

Kita melakukan dekompresi pada data **sdb.bak**, lalu memberikan output hasil dekompresi tersebut kepada **dd**, yang akan memasukan data tersebut ke dalam disk **/dev/sdb**. Dengan begitu data sudah kembali seperti semula tepat seperti saat data dibackup.

```
root@serverone:~# ls /usr
bin              include          index_hello.html  lib64  sbin         share
bye.index.html   index2.html      index.html        local  scoutlegion  src
games            index_bye.html   lib               root   sdb.bak
root@serverone:~# gzip -cd /usr/sdb.bak | dd of=/dev/sdb
131072+0 records in
131072+0 records out
67108864 bytes (67 MB) copied, 16.6261 s, 4.0 MB/s
root@serverone:~# _
```

Dengan menggunakan cara seperti ini, maka keamanan data bisa terjamin. Usahakan untuk selalu menggunakan media eksternal untuk dijadikan sebagai media penyimpanan hasil backup. Lalu simpan media tersebut di tempat yang aman.

23. **Pengamanan Data Backup**

Data yang sudah dibackup bukan berarti kebal dari segala macam gangguan. Justru yang harus dilakukan adalah menjaga data backup tersebut aman dan jelas. Apabila data perusahaan penuh dengan rahasia perusahaan dibackup, lalu hasil backup itu tercuri maka akan merugikan perusahaan dalam jumlah yang besar.

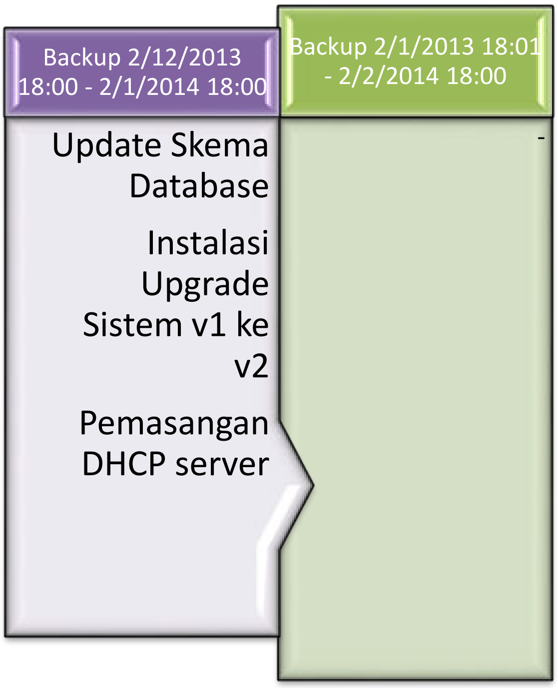
Karena itu, pengamanan data backup harus dilakukan sebaik mungkin, secara fisik maupun logik. Apabila memungkinkan, digunakan ruangan khusus untuk menyimpan media backup dilengkapi dengan keamanan dari api, air, suhu, penyusup dan berbagai macam pengganggu lainya. Lengkapi dengan peralatan keamanan seperti kapak, pemadam api, kamera CCTV, sensor panas, dsb.

Apabila data tersebut bisa dilakukan enkripsi, maka lakukan ekripsi terhadap data tersebut dan pastikan bahwa enrkipsi tersebut menjamin keaslian dan keamanan data.

Dengan proses enkripsi, akan memakan waktu dan proses dua kali, tapi keamanan akan terjaga dan meskipun ada pencurian, tidak akan bisa membuka rahasia dari perusahaan.

Penyimpanan dari hasil backup juga harus dilakukan serapi dan sejelas mungkin, sehingga apabila terjadi apa-apa tidak perlu mencari-cari data backup mana yang harus direcovery.

Membuat sebuah rak penyimpanan juga dianjurkan, di mana ditata media-media backup tersebut dan diberi label berdasarkan tanggal saat dilakukan backup terakhir hingga saat backup tersebut terjadi. Apabila jelas inti dari perubahan data dalam jangka waktu itu, berikan informasi jelas tentang apa saja yang berubah dalam rentang waktu backup tersebut.



Dengan membentuk informasi backup yang jelas, maka penelusuran masalah dan juga recovery data akan berlangsung cepat dan jelas bagian mana yang bertanggung jawab atas berbagai perubahan yang terjadi.

24. **Menggunakan Cloud Sebagai Penyimpanan**

Apabila digunakan aplikasi seperti **duplicity** yang bisa langsung mengunggah data backup ke cloud. Maka hal yang harus diperhatikan adalah keamanan dan kehandalan dari server cloud yang dipakai.

Ingat cloud mempunyai keterbatasan, salah satu keterbatasanya adalah kecepatan jaringan. Apabila jaringan tersebut lambat, maka proses backup yang menggunakan disk berukuran ratusan GB akan sangat merepotkan. Disamping proses backup, proses recovery juga akan memakan waktu yang lama.

Jadi, sebelum memutuskan untuk menggunakan cloud, lihat dulu infrastruktur sistem. Apakah sudah mendukung untuk menggunakan jaringan internet, atau cloud sebagai media penyimpananya.

Selain kecepatan, lihat apakah penyedia penyimpanan cloud tersebut menjamin kerahasiaan data. Apabila tidak, lakukan enkripsi baru kirimkan data ke cloud. Lihat apakah penyedia cloud tersebut mempunyai waktu operasi tanpa henti, dan lihat apakah data bisa aman dan tidak terjadi korupsi di penyimpanan.

Penyimpanan cloud biasanya sangat terbatas, apabila ada server dengan ukuran hardisk 16 TB, maka saat melakukan backup ke cloud, pasti akan membutuhkan proses yang lama, serta menghabiskan biaya yang cukup besar untuk menyewa ruang penyimpanan yang besar.

Salah satu keunggulan cloud adalah dia bisa diakses dari manapun, selama ada koneksi internet. Jadi meskipun mempunyai server yang mobile, maka bisa dilakukan backup dengan fleksibel dimanapun, kapanpun.

a. Rangkuman

Data merupakan sesuatu yang sangat penting dan salah satu alasan adanya komputer adalah untuk mengolah data. Data harus bisa diselamatkan dalam kondisi bagaimanapun dan kapanpun, oleh karena itu manajemen proses

penyelamatan dan pengembalian data harus dilakukan, dikenal dengan nama *backup and recovery process.*

Ketika ada data yang krusial, penting, maka keberadaan, kelangsungan, dan keamanan data menjadi prioritas utama. Karena itulah diciptakan aplikasi yang bisa mengatur keberadaan, kelangsungan, dan kemanan data.

Aplikasi ini akan menyalin data yang penting, lalu dipindahkan ke media lainya sebagai cadangan yang sewaktu-waktu data salinan ini bisa dikembalikan ke media aslinya dalam bentuk sama persis seperti saat penyalinan.

Faktor-faktor yang mempengaruhi kelangsungan data seperti usia data, usia media, kerusakana media, virus, pencurian, dsb, merupakan alasan utama bahwa backup data secara berkala harus dilakukan untuk melindungi data.

Data bisa dibackup berdasarkan klasifikasi, atau bisa langsung dibackup keseluruhan disk. Cara yg paling praktis adalah melakukan backup satu disk penuh, karena tidak menghabiskan waktu untuk memilah-milah mana data yang penting.

Apalagi saat recovery, harus kembali menempatkan data-data tersebut pada tempatnya. Apabila menggunakan disk, maka sekali disk di backup, maka dengan satu recovery pula seluruh disk akan kembali.

Pada sistem operasi Linux Debian, ada sebuah program yang bernama Dataset Definition, yang digunakan untuk melakukan backup satu disk penuh yang bisa disimpan dalam bentuk file atau langsung ke media penyimpanan lainya.

Setelah data di backup dengan sempurna, maka penyimpanan hasil backup harus diperhatikan. Data yang sensitif diincar banyak orang, pencurian terhadap data bisa terjadi. Salah satu usaha untuk meminimalisir ini adalah melakukan enkripsi terhadap data, dan menempatkan data di tempat yang aman.



Aman tidak hanya dari jangkaun manusia, tapi juga dari kecelakaan seperti kebakaran, banjir, suhu yang ekstrim, juga kerusakan fisik seperti jatuh atau dihancurkan oleh binatang.

Penempatan tatanan backup juga harus diperhatikan, tidak bisa serta merta menempatkan hasil backup di gudang dan dibiarkan usang, susah dicari dan berharap backup tidak pernah terjadi.

Penempatan yang jelas, pelabelan, dan informasi seputar backup bisa meminimalisir waktu yang diperlukan untuk proses backup. Sehingga data bisa lebih cepat dicari, lalu melakukan recovery.

Selain media penyimpanan lokal, bisa juga menggunakan jasa penyimpanan cloud yang menjamin keberadaan data. Namun, memilih penyedia jasa penyimpanan cloud tidak bisa sembarangan mengingat data yang dibackup itu besar dan penting.

Integritas, kecepatan, dan juga kehandalan penyimpanan cloud harus dipertimbangkan. Seberapa cepat data bisa dibackup, sebarap cepat data bisa direcover, amankah jaringan saat proses backup dan recovery, apakah akan terjadi kegagalan saat proses?

Hal tersebut harus diperhatikan untuk menjamin bahwa data yang ada tersimpan aman, dimanapun, dan kapanpun.

b. Tugas

1. Apa tujuan dari backup dan recovery?
2. Apa alasan backup dilakukan secara terpisah?
3. Apa alasan backup dilakukan secara utuh?
4. Apa yang menjadi alasan utama dilakukan backup?
5. Apa fungsi dari aplikasi `Dataset Definition`?

6. Sebutkan langkah-langkah dalam melakukan backup menggunakan perintah dd!
7. Mengapa lebih baik hasil backup dikompres?
8. Sebutkan berbagai macam upaya untuk mengamankan media backup!
9. Manakah yang lebih cocok sebagai media penyimpanan? Cloud atau media local, kemudian jelaskan!
10. Sebutkan ciri-ciri data yang harus dibackup, dan kenapa harus dilakukan backup!

c. Test Formatif

1. Apa itu backup dan recovery?
   a) Proses mengembalikan data
   b) Proses menyalin data untuk dikembalikan sewaktu-waktu
   c) Proses menyalin data untuk diamankan
   d) Proses menyalin data di jaringan internet
   e) Salah semua
2. Apabila media penyimpanan dan waktu cukup, backup seperti apa yang seharusnya dilakukan?
   a) Backup satu disk langsung ke disk lainya
   b) Backup satu disk ke cloud
   c) Backup terpisah-pisah ke disk lainya
   d) Backup terpisah-pisah ke cloud
   e) Semua salah
3. Waktu optimal untuk melakukan backup pada sistem yang biasa, tidak begitu krusial adalah ..
   a) 1 Hari
   b) Hari
   c) 1 Minggu
   d) Minggu
   e) 1 Bulan
4. Yang bukan penyebab kerusakan data adalah ...
   a) Kegagalan hardware
   b) Virus

c) Salah Hapus
d) Penyalinan
e) Semua salah

5. Aplikasi bawaan Linux Debian untuk melakukan backup adalah ...
   a) Backloud
   b) DD
   c) Backup Vi
   d) CL
   e) G++
6. Direktori tempat virtualisasi dari hardware di Linux ada di ...
   a) /usr
   b) /bin
   c) /home
   d) /dev
   e) /etc
7. Data yang dibackup harus diamankan lagi, cara yang tepat untuk mengamankan media penyimpanan data setelah dilakukan backup adalah ...
   a) Mengenkripsi data
   b) Memberikan kata kunci untuk bisa membaca media
   c) Menempatkan media penyimpanan di tempat yang aman
   d) Jawaban A dan B benar
   e) Salah semua
8. Apakah yang membuat media penyimpanan local lebih dipilih sebagai media utama untuk backup?
   a) Letak, Keamanan, Pengamanan bisa dilakukan secara transparan dan jelas
   b) Ukuran, Kecepatan, dan Keefektifan sangat tinggi
   c) Akses, Penggunaan, Recovery bisa dilakukan di satu tempat
   d) Jawaban A dan C benar
   e) Jawaban A dan B Benar
9. Yang bukan keunggulan dari media penyimpanan cloud adalah ...
   a) Bisa diakses dari manapun
   b) Keberadaan dijamin oleh penyedia

c) Bebas gangguan jaringan

d) Semua benar

e) Semua salah

10. Yang bukan keunggulan dari media penyimpanan local adalah ...

a) Akses hanya dari satu tempat

b) Keberadaan terjamin dan transparan oleh badan pengaman

c) Bebas gangguan jaringan

d) Semua benar

e) Semua salah

d. Lembar Jawaban Tes Formatif

e. Lembar kerja Siswa

14. Kegiatan Belajar 2 : Menyajikan Hasil Manajemen Backup dan Recovery Linux.

Tujuan Pembelajaran

Setelah mengikuti kegiatan belajar 2 ini, siswa diharapkan dapat:

i. Menyajikan hasil manajemen backup dan recovery di sistem operasi Linux Debian dalam bentuk hasil manajemen backup, laporan, makalah, dan presentasi.

a. Uraian Materi

ii. Penyajian Hasil Manajemen Backup dan Recovery.

Siswa mendapat tugas untuk membuat kelompok yang terdiri dari 3 – 5 orang. Setiap kelompok akan melakukan praktik membuat sebuah backup dalam bentuk disk menjadi file backup.

Setelah dilakukan praktikum backup dan recovery, maka siswa ditugaskan untuk membuat laporan tentang langkah-langkah dan kesalahan yang ditemui dan cara pemecahan masalah tersebut.

Lalu membuat makalah tentang backup dan recovery di sistem besar (perusahaan) lalu melakukan presentasi berdasarkan makalah tersebut yang sudah dikompakan untuk presentasi yang padat, ringkas, dan cepat.

b. Rangkuman

c. Tugas

1. Lakukan sebuah proses backup recovery pada perangkat masin-masing dimana dilakukan backup dari satu disk menjadi sebuah file.
2. Buat laporan tentang kesalahan yang terjadi, dan cara penangananya apabila terjadi selama proses backup.
3. Buat makalah tentang backup dan recovery di sistem besar (perusahaan), tentang seberapa sering periode backup, apa saja yang dibackup, media penyimpanan apa yang cocok, serta bagaimana cara mengatur media penyimpanan tersebut.
4. Sajikan makalah yang sudah diolah dalam bentu presentasi.

d. Tes Formatif

e. Lembar Jawaban Tes Formatif

f. Lembar Kerja Siswa

15. Kegiatan Belajar 1 : Memahami Manajemen Remote Access

b. Tujuan Pembelajaran

Setelah mengikuti kegiatan belajar 1 ini, siswa diharapkan dapat :

i. Memahami cara memanejemen remote access.

c. Uraian Materi

i. Remote Access

1. Pengertian Remote Access

Remote access yang berarti akses dari jarak jauh merupakan suatu cara untuk menajemen server tanpa harus secara fisik menyentuh server dan melakukan manajemen di daerah operasional server. Remote access berguna

untuk membuat pekerjaan bisa dilakukan di mana saja dan kapanpun selama ada akses jaringan ke server.

Remote access bisa dicapain dengan berbagai cara, yang pasti server yang di remote harus bisa diakses oleh host yang bersangkutan baik melalui jaringan internet publik atau menggunakan VPN.

Salah satu remote access yang aman adalah menggunakan SSH (Secure Shell). Seperti namanya, SSH menyediakan koneksi untuk melakukan remote dengan aman dengan interface command line meskipun dengan jaringan yang tidak aman.

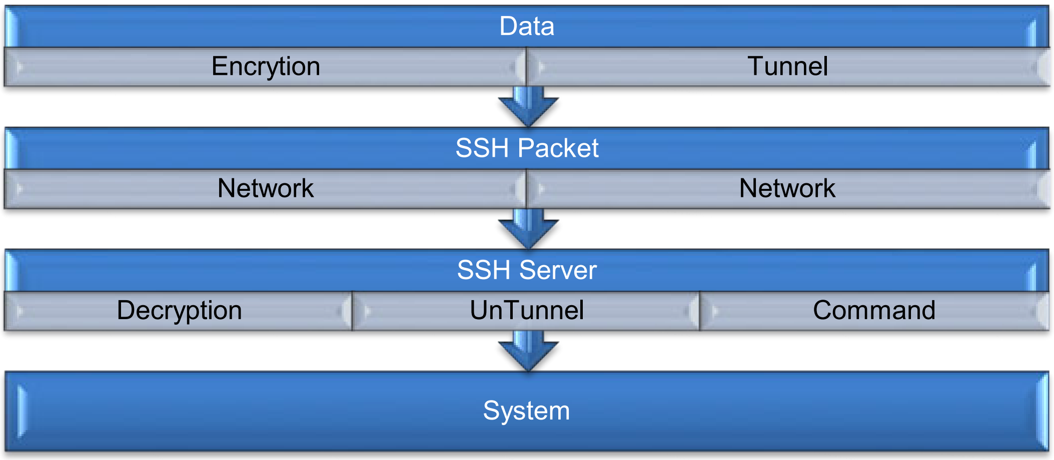
Aplikasi dari SSH ini biasanya digunakan untuk login sistem UNIX, untuk menggantikan sistem remote seperti telnet yang mengirim informasi password dengan tulisan biasa tanpa enkripsi.

2. **SSH**

SSH berjalan dengan dua basis, satu sebagai server, dan satu sebagai client. SSH menggunakan kriptografi publik untuk melakukan autentikasi pengguna. SSH akan membuat *public-private key pairs* yang digunakan untuk autentikasi dan enkripsi, yaitu sebuah kunci pasangan yang akan digunakan untuk autentikasi pengguna. Setelah itu pengguna bisa memasukkan kata sandi untuk masuk.

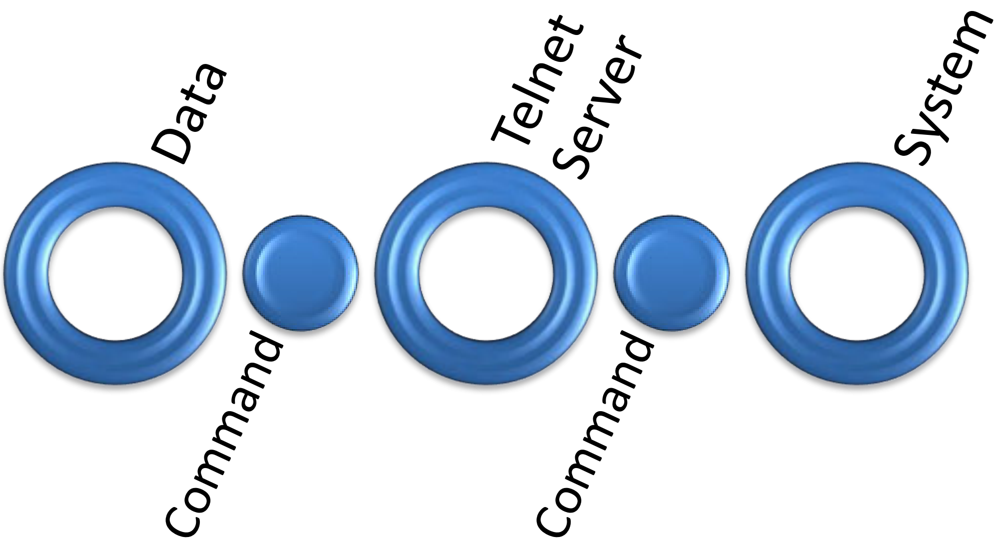
Cara lainya adalah SSH akan menggunakan kunci pasangan yang harus di pasang secara manual. Dengan cara ini membutuhkan waktu yang lama, dan SSH akan melakukan pengecekkan apakah pengguna dengan kunci publik yang diberikan mempunyai kunci pribadi yang sesuai. Tapi kata sandi sudah tidak dibutuhkan lagi, dengan ini harus lebih berhati-hati dalam autentikasi kunci publik yang tidak dikenal.

SSH biasanya digunakan untuk masuk secara remote ke server lalu melakukan ekseksui perintah-perintah secara remote. Tapi SSH juga mendukung tunneling, meneruskan port TCP, melakukan transfer file, pengkopian file. SSH menggunakan model client-server.



Alur data SSH, dari data yang mungkin saja itu berupa command. Akan dienkripsi, tunneling, lalu diatur menjadi paket SSH. Berjalan melalui jaringan-jaringan, dan masuk ke SSH server untuk dibuka lagi paketnya, di decrypt, dan membuka semua isi data dan berubah jadi perintah yang akhirnya diberikan kepada system.

Berbeda dengan alur data Telnet biasa, yang berbentuk seperti ini.



Di mana tidak ada proses penyembunyian data, sehingga data sensitif bisa ditangkap oleh pihak yang mempunyai akses ke jaringan ini.

Dengan menggunakan SSH, kita bisa masuk ke dalam server dari jarak jauh, melewati koneksi yang tidak aman, dan tetap mendapat jaminan keamanan dari SSH.

Secara default, SSH server berjalan di atas port 22. Port ini bisa dirubah sesuai kebutuhan dan biasanya dirubah untuk kamuflase yang membuat orang mengira tidak ada SSH server di server tersebut.

3. **SSH Linux Debian**

Sekarang kita akan mencoba untuk memasang server SSH di server kita. Sehinga kita tidak perlu lagi mengakses secara langsung, cukup menggunakan SSH untuk mengakses server dari komputer lain, bahkan dari komputer dengan OS selain Linux.

Gunakan **apt** untuk install server SSH di server apabila di server belum ada SSH server. Untuk mengecek apabila SSH server sudah ada atau belum. Lihat apakah ada direktori **/etc/ssh** di server kita. Apabila ada, berarti SSH server sudah terinstal.

Apabila server sudah terinstall, maka akan ada direktori tersebut dan di dalamnya ada struktur seperti berikut.



Seperti biasa, karena kita adalah server, maka kita berfungsi sebagai *daemon*, sehingga kita harus merubah konfigurasi daemon SSH kita, yang berada di **sshd_config** (SSH Daemon Config).

Secara default, SSH server sudah terpasang dengan baik, dan kita sudah bisa melakukan remote access tanpa perlu merubah satu konfigurasi pun. Cara yang paling baik untuk memulai adalah mengatur agar IP server kita bisa dicapai oleh host lain yang terhubung dengan IP server. Dalam kasus ini IP server di interface yang terhubung dengan PC Windows adalah **eth1** dengan IP 192.168.2.1

Lakukan konfigurasi interface dengan mengedit file **/etc/network/interfaces** dengan **nano**. Lalu rubha hingga interface **eth1** yang terhubung dengan PC Windows dengan kabel RJ45 mempunyai IP 192.168.2.1 dengan netmask 255.255.255.0.

```
# and how to activate them. For more information, see interfaces(5).

# The loopback network interface
auto lo
iface lo inet loopback

# The primary network interface
allow-hotplug eth0
iface eth0 inet dhcp

auto eth1
iface eth1 inet static
address 192.168.2.1
netmask 255.255.255.0
~
~
~
~
~
~
~
~
~
"/etc/network/interfaces" 15L, 353C written
root@serverone:/etc/ssh#
```

Setelah itu, pindah ke PC Windows, lakukan konfigurasi di PC Windows untuk mempunyai IP 192.168.2.2 dengan netmask 255.255.255.0.

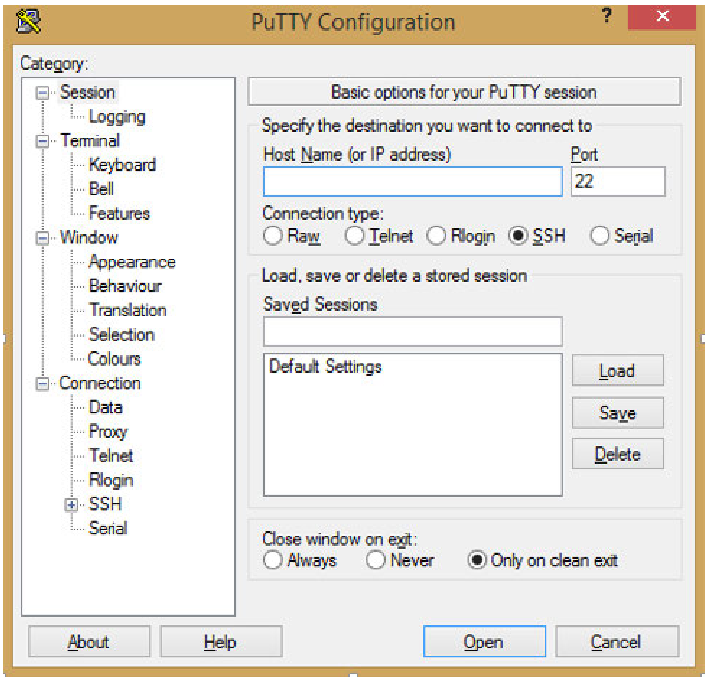
Setelah melakukan perubah konfigurasi interface, jangan lupa untuk merestart peralatan networking dengan cara.

```
/etc/init.d/networking restart
```

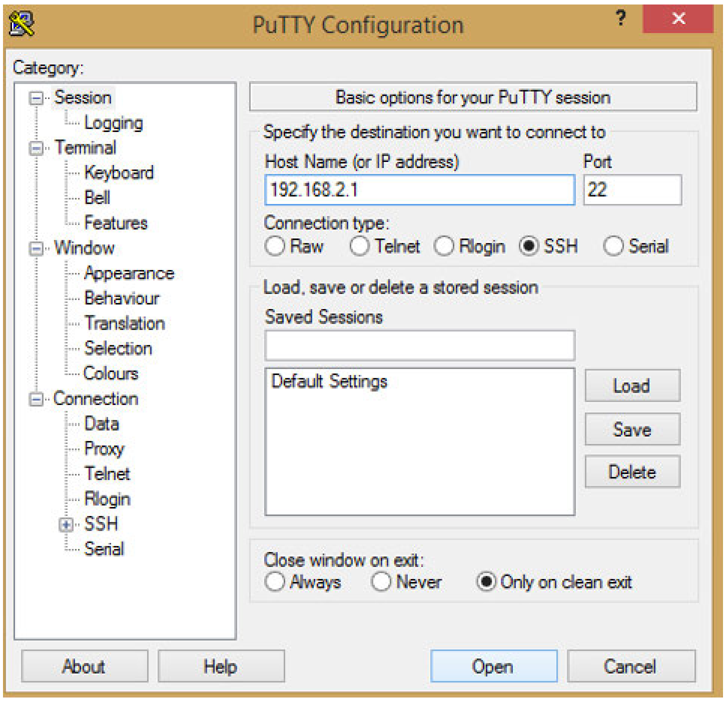
Tunggu beberapa saat hingga selesai.

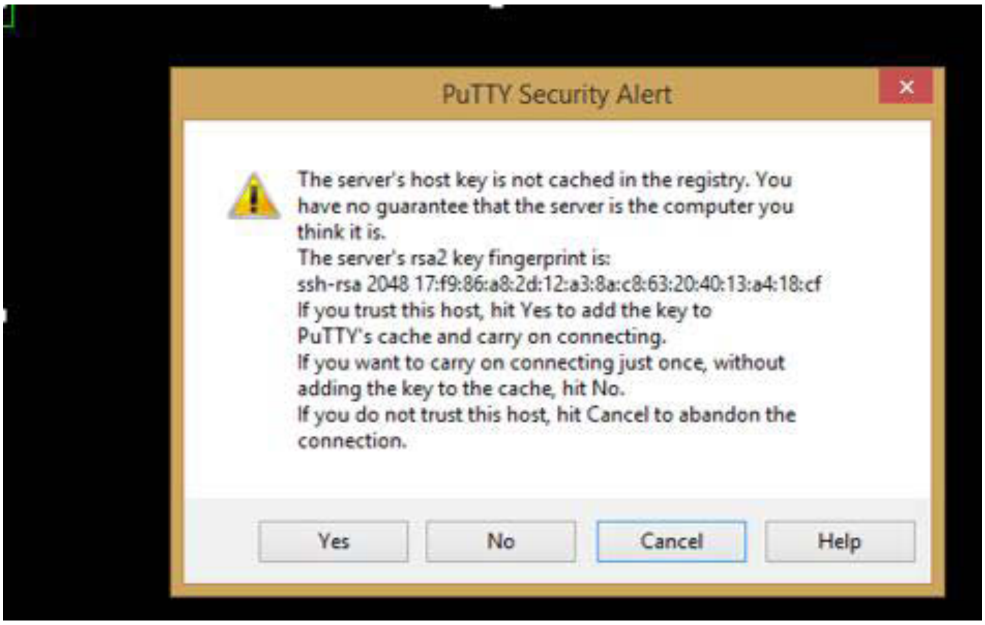
Untuk melakukan *remote secure shell* di Windows, kita menggunakan software yang bernama Putty (bisa diunduh di http://ww.putty.org).

Ini tampilan dari PuTTY yang berjalan di PC Windows.

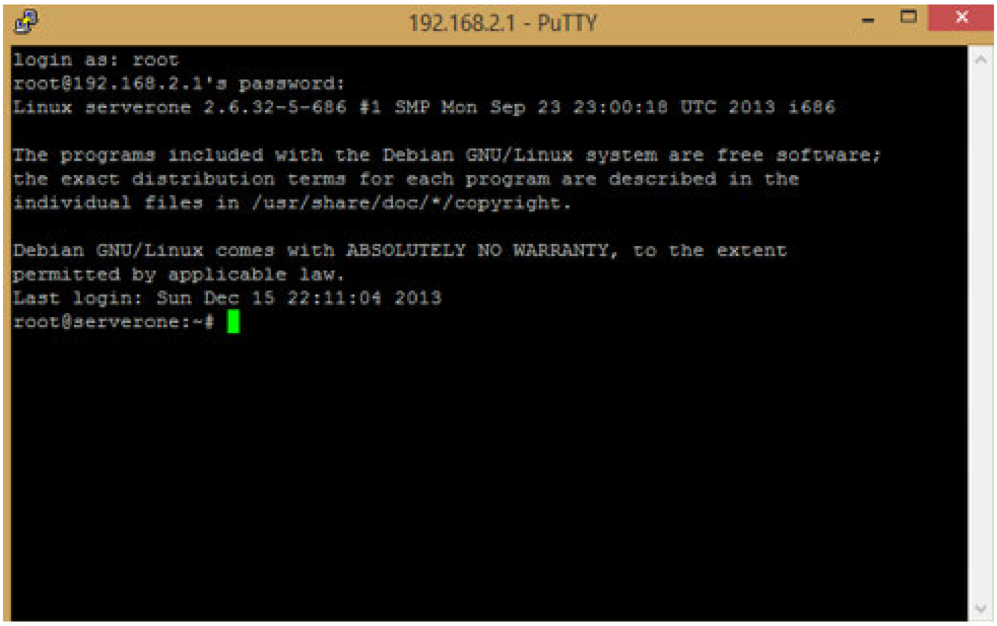


Karena kita sudah terhubung dengan server dalam satu jaringan yang sama, dalam kasus ini IP PC Windows adalah 192.168.2.2, dan server adalah 192.168.2.1. Maka untuk melakukan koneksi ke server, kita memasukkan IP dari server dan klik tombol **Open.**





PuTTY akan memperingatkan bahwa server ini belum termasuk server yang dikenal karena kita baru pertama ini melakukan koneksi. Pilih **Yes**, kita tahu bahwa ini server yang kita mau.

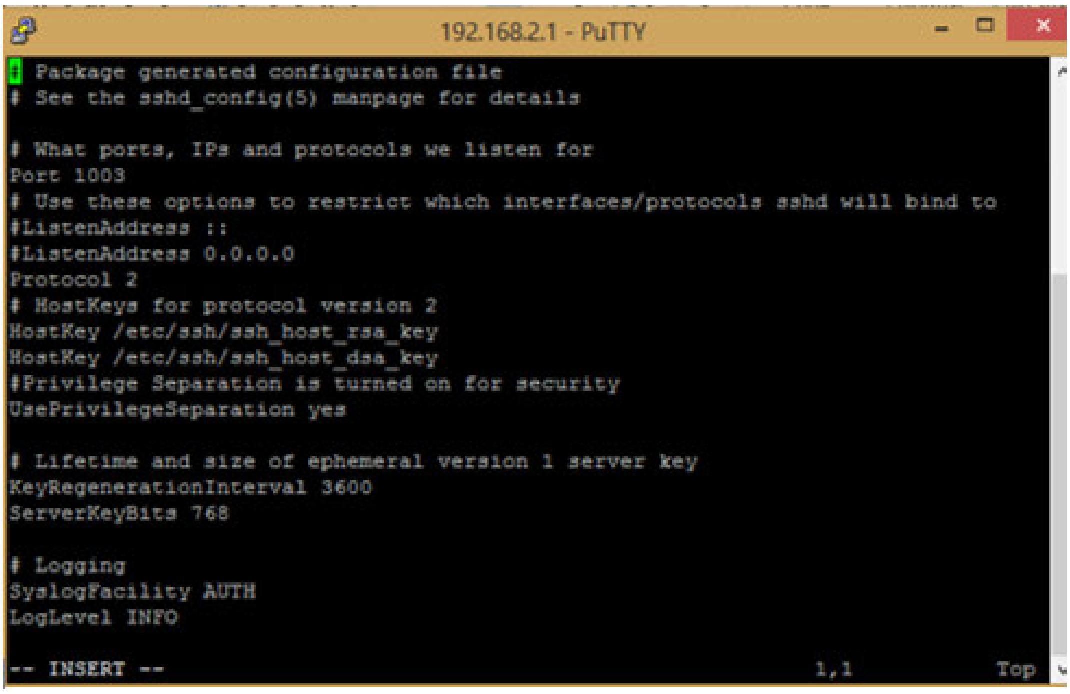


Maka akan ada prompt yang menanyai tentang identitas yang kita gunakan untuk masuk ke dalam server. Kita gunakan identitas **root** dengan kata sandi **root.**

Sekarang kita sudah masuk ke sistem server kita sebagai **root** secara remote menggunakan SSH. Apa saja yang bisa kita lakukan? Kita akan mencoba merubah port SSH server secara remote.

Gunakan text editor **nano** untuk merubah isi dari **/etc/ssh/sshd_config**.

Rubah port tempat SSH server berjalan di port 1003.




```
# Package generated configuration file
# See the sshd_config(5) manpage for details

# What ports, IPs and protocols we listen for
Port 1003
# Use these options to restrict which interfaces/protocols sshd will bind to
#ListenAddress ::
#ListenAddress 0.0.0.0
Protocol 2
# HostKeys for protocol version 2
HostKey /etc/ssh/ssh_host_rsa_key
HostKey /etc/ssh/ssh_host_dsa_key
#Privilege Separation is turned on for security
UsePrivilegeSeparation yes

# Lifetime and size of ephemeral version 1 server key
KeyRegenerationInterval 3600
ServerKeyBits 768

# Logging
SyslogFacility AUTH
LogLevel INFO

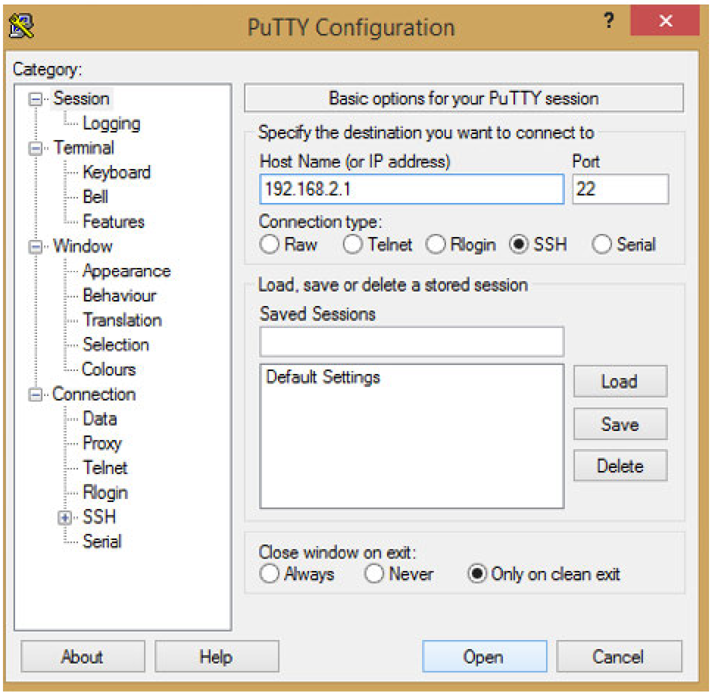
-- INSERT --                                              1,1           Top
```

Port SSH server yang semula 22, dirubah menjadi 1003. Setelah melakukan perubahan, simpan file, lalu restart SSH server dengan perintah ini.

```
/etc/init.d/ssh restart
```

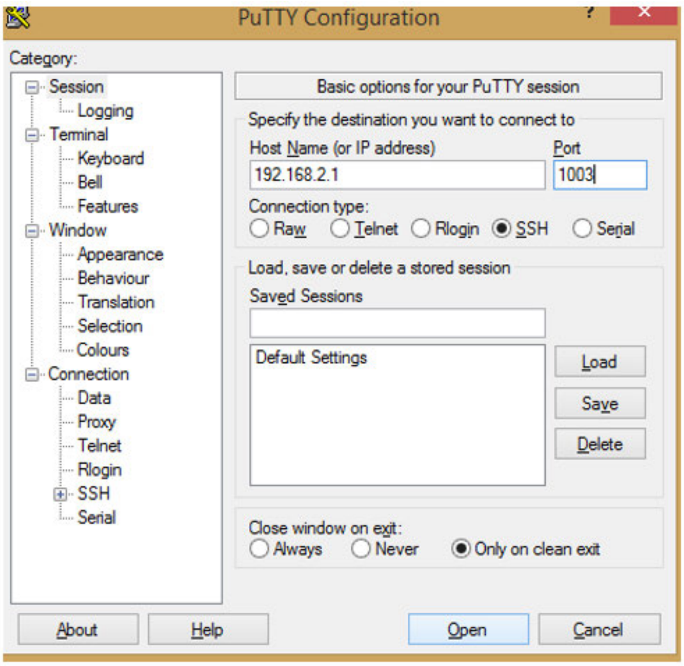
Tunggu beberapa saat hingga SSH server selesai merestart. Lalu tutup window dari PuTTY.

Coba lagi buka PuTTY, dan masukkan informasi sama seperti waktu masuk pertama.



Akan ada pesan bahwa koneksi ditolak. Itu berarti kita telah berhasil merubah port dari SSH server yang ada di server yang baru saja kita remote dan merubah port dari SSH server.

Coba lagi, kali ini ganti port 22 menjadi 1003.



Pilih **Open**, maka akan ada peringatan lagi, pilih **Yes**. Maka kita akan dihadapkan dengan prompt login seperti tadi.

Dengan ini, berarti kita sudah bisa mengakses dan mengatur server kita secara remote menggunakan SSH. Tidak peduli apakah kita menggunakan Windows, Mac, Linux, selama support dengan SSH, maka server kita bisa kita kendalikan dengan bagaimanapun, kapanpun, dan dimanapun.

d. Rangkuman

Remote Access merupakan suatu cara untuk melakukan akses ke suatu sistem secara jarak jauh atau remote. Remote access berguna apabila kita menghendaki akses ke suatu sistem tanpa harus berada di hadapan sistem tersebut secara fisik.

Kita bisa mengakses suatu sistem dari jarak yang jauh, melewati jaringan yang terhubung dan mencapai sistem secara aman menggunakan sebuah metode enkripsi dan autentikasi. Kesatuan sistem pengaksesan remote ini sudah terimplementasi dengan adanya SSH, Secure Shell.

Secure shell merupakan metode akses remote yang aman, meskipun dia berjalan di atas jaringan yang tidak aman dengan menerapkan enkripsi kunci publik. Dengan ini maka keamanan bisa terjamin.

SSH berjalan di atas port 22, dan menerapkan model client-server. Di mana client yang bertugas sebagai pengakses dari sistem yang diremote. SSH tidak membatasi berapa jumlah pengguna yang mengakses sistem, tapi mungkin sistem bisa membatasinya.

Sistem operasi Debian bisa menggunakan **openssh-server** sebagai aplikasi yang bisa diinstall menggunakan **apt-get** untuk dijadikan SSH server. Ketika server sudah terinstal, maka sudah bisa dilakukan akses remote dengan menggunakan software yang mendukung SSH.

Di sistem operasi Windows, bisa menggunakan PuTTY untuk mengakses SSH. Setelah mengakses dengan benar, maka akan muncul prompt login dengan nama pengguna dan kata sandi. Setelah masuk, maka semua yang bisa dilakukan di server secara langsung juga bisa dilakukan melalui komputer dengan sistem operasi Windows dari tempat yang berbeda, waktu yang berbeda.

e. Tugas

1. Apa yang dimaksud dengan remote access?
2. Apa yang membuat SSH aman?
3. Apa saja yang bisa dilakukan dengan SSH?
4. Sebutkan langkah-langkah installasi SSH server sampai bisa diremote lewat PC lainya!
5. Sebutkan cara merubah port default SSH menjadi 1003!
6. Kenapa SSH digunakan untuk remote access?
7. Apa kegunaan dari remote access?
8. Kenapa remote access digunakan, apa solusi lain dari remote access?
9. Apa beda remote access dengan direct access?
10. Bagaimana proses SSH menyampaikan data dari SSH client sampai SSH server? Jelaskan!

f. Test Formatif

i. Yang dimaksud remote access adalah ...
   a) Akses komputer dari komputer lain
   b) Akses komputer dari tempat lain
   c) Akses komputer tanpa perantara
   d) Semua salah
   e) Semua benar

ii. Remote access yang aman menggunakan ...
   a) SSP
   b) SSC
   c) SSH
   d) SSHD
   e) VPN



iii. SSH membuat sebuah kunci ganda yang bisa digunakan untuk autentikasi, bagaiamana cara autentikasi apabila tidak ada kunci ganda ?
   a) Menggunakan kunci utama
   b) Menggunakan kata sandi
   c) Menggunakan kunci kedua
   d) Menggunakan akses belakang
   e) Menggunakan SSH Tunneling

iv. Apa guna dari tunneling?
   a) Memasukkan data ke dalam paket terowongan
   b) Memasukkan data ke dalam paket lubang
   c) Membungkus data dengan keamanan dan enkripsi
   d) Membungkus data menuju paket enkripsi
   e) Semua salah

v. File yang berisi konfigurasi dari SSH adalah ...
   a) sshd_dhcp
   b) sshd_conf
   c) sshd_configs
   d) sshd_config
   e) ssh_config

vi. Akses SSH dari Windows bisa menggunakan software ...
   a) PuTTI
   b) SSHSuper
   c) SSHPy
   d) PuTTY
   e) SSHClientWindow

vii. Perbedaan remote dan direct access adalah ...
   a) Beberapa perintah tidak bisa dijalankan
   b) Tidak bisa login sebagai root
   c) Kemanan tidak terjamin
   d) Semua Benar
   e) Semua Salah



viii. Port default untuk SSH server adalah ...

a) 21

b) 22

c) 23

d) 24

e) 20

ix. Perintah untuk merestart SSH server adalah ...

a) /etc/init.d/sshd restart

b) /etc/init.d/open-ssh restart

c) /etc/init.d/ssh restart

d) /etc/ssh restart

e) /etc/sshd restart

x. Salah satu aplikasi SSH server di Linux adalah ...

a) SSH-Server

b) OpenSSH-server

c) SSHD-Server

d) SSH-Open-Server

e) SSHD-Open

g. Lembar Jawaban Tes Formatif

h. Lembar Kerja Siswa



16. Kegiatan Belajar 2 : Menyajikan Hasil Manajemen Remote Access

a. Tujuan Pembelajaran

Setelah mengikuti kegiatan belajar 2 ini, siswa diharapkan dapat:

i. Menyajikan hasil manajemen remote access dalam bentuk praktek, laporan, makalah, dan presentasi.

b. Uraian Materi

ii. Penyajian Hasil Manajemen Remote Access.

Siswa ditugaskan untuk mencoba mengimplentasikan remote access terhadap sebuah server, dengan menggunakan SSH yang setelahnya akan diakses menggunakan sebuah PC yang terhubung dengan server tersebut melalui media kabel.

Setelah itu siwa membuat laporan langkah-langkah yang tepat, dan apa saja kendala yang dihadapi saat melakukan remote access, dan apa keuntungan dan kekurangan menggunakan remote access.

Lalu siswa membuat makalah tentang implementasi remote access di kehidupan nyata, juga tentang penerapan SSH di salah satu bidang, lalu menyajikanya dalam bentuk presentasi.

c. Rangkuman

d. Tugas

1. Siswa membuat kelompok yang terdiri dari 3 – 5 orang.
2. Siswa melakukan implementasi remote access terhadap server dengan PC lainya yang terhubung dengan kabel.
3. Siswa membuat laporan dari proses implementasi remote access, tentang kendala, keuntungan dan kekurangan remote access.
4. Siswa membuat makalah tentang implementasi remote access di kehidupan nyata, juga penerapan SSH di salah satu bidang (Seperti SCP, Collaboration, dsb).
5. Hasil makalah diolah untuk dijadikan presentasi yang singkat, padat dan jelas.

e. Tes Formatif

f. Lembar Jawaban Tes Formatif

g. Lembar Kerja Siswa

17. Kegiatan Belajar 1 : Memahami Cara Mengkonfigurasi DHCP Server

a. Tujuan Pembelajaran

Setelah mengikuti kegiatan belajar 1 ini, siswa diharapkan dapat :

1. Memahami cara mengkonfigurasi DHCP server.

b. Uraian Materi

DHCP Server

1. Pengertian DHCP Server

DHCP (Dynamic Host Control Protocol) adalah protokol pengalamatan host secara dinamis. Dalam sebuah jaringan yang besar, akan ada bagian yang pengalamatan IP address tidak begitu kritikal. Di bagian ini pengalamatan IP bisa dilakukan secara dinamis dan otomatis.

Apabila dalam sebuah jaringan diwajibkan memberi IP satu per satu dengan manual, maka akan memakan waktu yang sangat lama. Misalkan ada jaringan dengan pengguna 1500 orang, maka akan membutuhkan pengaturan alamat IP secara manual di tiap komputer sebanyak 1500 kali.

Karena itulah DHCP ada, sehingga komputer host tetap bisa terhubung dengan jaringan secara otomatis meskipun tidak mendapatkan IP address sesuai yang diminta, tapi sudah pasti akan mendapatkanya apabila IP masih tersedia dan DHCP server berjalan normal.



Pendapatan IP mempunyai waktu yang terbatas, DHCP mengatur agar IP bisa digunakan berulang-ulang. Ada batas penyewaan waktu yang harus disetujui oleh host. Jadi ketika waktu penyewaan habis, maka host bisa menentukan apakah dia ingin menyewa IP lagi atau berhenti supaya DHCP server bisa memberikan IP tersebut ke host lainya.

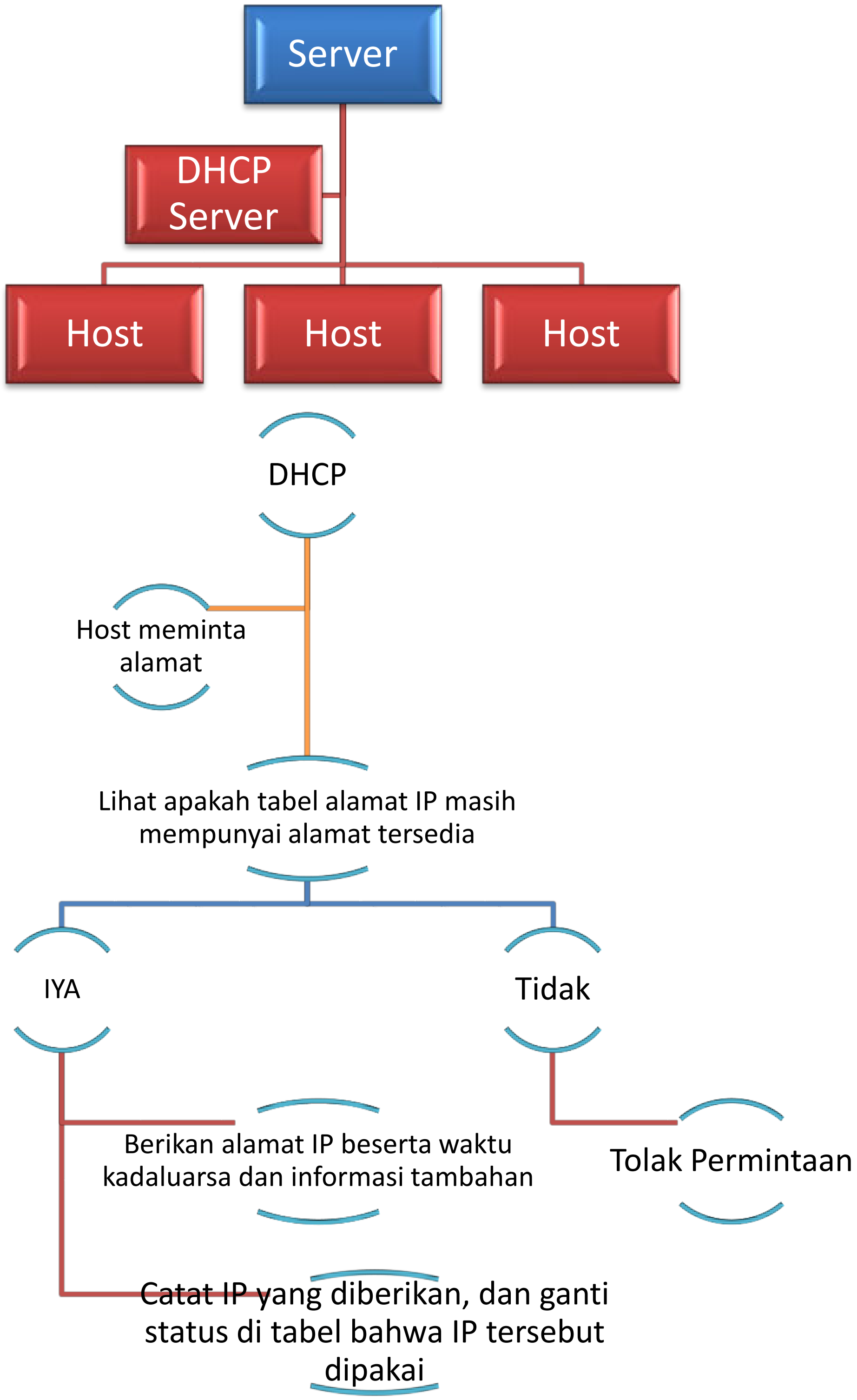
Beberapa IP juga bisa diberikan secara statis untuk MAC address tertentu. Sehingga IP tersebut bisa diserahkan secara ekslusif untuk beberapa mesin yang memang krusial dengan IP tersebut, misalnya membuat DNS server atau HTTP server local di daerah yang diatur IP nya oleh DHCP. Jadi DHCP tidak terbatas hanya bisa memberikan IP secara dinamis dan tidak teratur. Beberapa bisa teratur sehingga membuat DHCP lebih fleksible dalam berbagai keadaan.

2. Cara Kerja DHCP Server

DHCP server bekerja dengan cara menawarkan diri sebagai DHCP server dan menawarkan IP kepada host yang terhubung. Host akan meminta alamat IP kepada DHCP, lalu DHCP server akan memeriksa apakah masih ada alamat yang tersedia, dan alamat apa saja yang tersedia itu.

Setelah diketahui adanya alamat yang tersedia. Maka DHCP server akan memberikan kepada host tersebut alamat tersebut, DHCP juga menyimpan informasi tambahan seperti DNS server yang harus digunakan, beserta default gatewaynya.

Alamat IP diberikan lengkap dengan informasi kapan dia kadaluarsa sehingga host bisa meminta lagi dan DHCP server bisa menyatakan alamat tersebut sudah bebas dan bisa digunakan kembali baik oleh host yang sama atau berbeda.


Server
DHCP Server
Host
Host
Host
DHCP
Host meminta alamat
Lihat apakah tabel alamat IP masih mempunyai alamat tersedia
IYA
Tidak
Berikan alamat IP beserta waktu kadaluarsa dan informasi tambahan
Tolak Permintaan
Catat IP yang diberikan, dan ganti status di tabel bahwa IP tersebut dipakai



DHCP server mempunyai batas dari IP mana sampai mana dia bisa memberikan alamat tersebut kepada host. Dengan batas ini jumlah host bisa dibatasi sesuai dengan keperluan. Digunakan sebagai alternatif untuk menjaga server dari koneksi host yang tidak diinginkan.

3. **Mesin DHCP Server**

   Biasanya, dalam suatu jaringan yang diatur oleh router sudah memiliki DHCP server sendiri di routernya. Namun, apabila harus menggunakan server seperti Linux Debian, maka kita harus memasang aplikasi yang bisa menjadikan server kita sebagai DHCP server.

   Di Linux Debian, aplikasi yang bisa digunakan sebaai DHCP server adalah **dhcp3-server**.

4. **DHCP Server Debian**

   DHCP server bisa diinstall dengan menggunakan perintah **apt-get install <nama_paket>**. Dalam kasus ini paket yang kita install bernama **dhcp3-server.**

```
apt-get install dhcp3-server
```

   Biarkan beberapa saat, apabila ada pertanyaan Y/n, tekan enter untuk mengijinkan installasi DHCP server. Ketika installasi bila ada tulisan **failed**, biarkan saja, karena kita memang belum melakukan konfigurasi



```
root@serverone:~# apt-get install dhcp3-server
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following NEW packages will be installed:
  dhcp3-server
0 upgraded, 1 newly installed, 0 to remove and 0 not upgraded.
Need to get 0 B/26.3 kB of archives.
After this operation, 65.5 kB of additional disk space will be used.
Selecting previously deselected package dhcp3-server.
(Reading database ... 49284 files and directories currently installed.)
Unpacking dhcp3-server (from .../dhcp3-server_4.1.1-P1-15+squeeze8_all.deb) ...
Setting up dhcp3-server (4.1.1-P1-15+squeeze8) ...
root@serverone:~# _
```

Apabila tidak ada tulisan **failed** juga tidak mengapa. Setelah selesai installasi, coba lihat apakah ada direktori **/etc/dhcp**. Lakukan **cd** terhadap direktori tersebut dan lakukan **ls** untuk melihat isinya.

```
root@serverone:~# cd /etc/dhcp/
root@serverone:/etc/dhcp# ls
dhclient.conf  dhclient-enter-hooks.d  dhclient-exit-hooks.d  dhcpd.conf
root@serverone:/etc/dhcp# _
```

Apabila kita ingin melakukan konfigurasi server DHCP, gunakan file yang bernama **dhcpd.conf**. Gunakan **nano** untuk merubah isi dari **dhcp.conf.**

```
  GNU nano 2.2.4              File: dhcpd.conf

#
# Sample configuration file for ISC dhcpd for Debian
#
#

# The ddns-updates-style parameter controls whether or not the server will
# attempt to do a DNS update when a lease is confirmed. We default to the
# behavior of the version 2 packages ('none', since DHCP v2 didn't
# have support for DDNS.)
ddns-update-style none;

# option definitions common to all supported networks...
option domain-name "example.org";
option domain-name-servers ns1.example.org, ns2.example.org;

default-lease-time 600;
max-lease-time 7200;

# If this DHCP server is the official DHCP server for the local
# network, the authoritative directive should be uncommented.

^G Get Help  ^O WriteOut  ^R Read File ^Y Prev Page ^K Cut Text  ^C Cur Pos
^X Exit      ^J Justify   ^W Where Is  ^V Next Page ^U UnCut Text^T To Spell
```

Di atas ini isi dari file **dhcp.conf**. Ada beberapa pengaturan seputar konfigurasi DHCP.

Yang perlu diperhatikan adalah bagian ini,

```
GNU nano 2.2.4                File: dhcpd.conf

#  option broadcast-address 10.254.239.31;
#  option routers rtr-239-32-1.example.org;
#}

# A slightly different configuration for an internal subnet.
#subnet 10.5.5.0 netmask 255.255.255.224 {
#  range 10.5.5.26 10.5.5.30;
#  option domain-name-servers ns1.internal.example.org;
#  option domain-name "internal.example.org";
#  option routers 10.5.5.1;
#  option broadcast-address 10.5.5.31;
#  default-lease-time 600;
#  max-lease-time 7200;
#}

# Hosts which require special configuration options can be listed in
# host statements.   If no address is specified, the address will be
# allocated dynamically (if possible), but the host-specific information
# will still come from the host declaration.

^G Get Help   ^O WriteOut   ^R Read File  ^Y Prev Page  ^K Cut Text   ^C Cur Pos
```

Di bagian ini kita akan merubah konfigurasi yang semula dimatikan, supaya aktif. Rubah hingga berbentuk seperti ini.

```
GNU nano 2.2.4                File: dhcpd.conf

#  option broadcast-address 10.254.239.31;
#  option routers rtr-239-32-1.example.org;
#}

# A slightly different configuration for an internal subnet.
subnet 192.168.1.0 netmask 255.255.255.0 {
  range 192.168.1.2 192.168.1.254;
  option domain-name-servers 192.168.1.1;
  option domain-name "server";
  option routers 192.168.1.1;
  option broadcast-address 192.168.1.255;
  default-lease-time 3600;
  max-lease-time 7200;
}

# Hosts which require special configuration options can be listed in
# host statements.   If no address is specified, the address will be
# allocated dynamically (if possible), but the host-specific information
# will still come from the host declaration.

                        [ Wrote 107 lines ]

root@serverone:/etc/dhcp# _
```

Kita membuat aturan dhcp untuk subnet 192.168.1.10, dengan netmask 255.255.255.0 atau 192.168.1.10/24. Lalu kita menentukan, bahwa IP yang bisa digunakan atau disewa oleh host adalah antara 192.168.1.2 – 192.168.1.254. Lalu kita setting alamat DNS server, yaitu mesin server sendiri 192.168.1.1, dengan domain name server. Lalu setting bahwa alamat

broadcastnya adalah 192.168.1.255, dengan waktu sewa default 3600 dan waktu sewa maksimal 7200 detik.

Setelah selesai mengatur konfigurasi DHCP server, kita perlu menentukan di bagian mana DHCP server kita ini akan berjalan. Ketikkan,

```
nano /etc/default/dhcp
```

Isi dengan,

```
INTERFACE=”eth1”
```

Apabila di server anda tidak ada interface ethernet 1, silahkan ganti menjadi interface ethernet yang ada di sistem anda. Ethernet 0 atau **eth0** biasanya sudah ada dan siap digunakan.

Di sini berarti DHCP server kita berjalan di atas **eth1**, jadi apabila ada host yang terkoneksi dengan **eth1**, maka host tersebut bisa meminta IP dari server kita.

Coba sambungkan PC dengan interface **eth1** di server, apabila menggunakan sistem oeprasi windows, maka buka command prompt dan gunakan **ipconfig** untuk melihat apakah sudah mendapatkan IP dari server kita. Untuk sistem Linux, gunakan **ifconfig**.

c. Rangkuman

DHCP server adalah server yang melayani protokol untuk penyewaan IP. Dalam sebuah jaringan yang besar mempunyai banyak pengguna, maka tanpa DHCP butuh dilakukan konfigurasi satu per satu yang sangat melelahkan.

Dengan adanya DHCP, maka konfigurasi otomatis bisa dijalankan. DHCP menyewakan IP dalam satu jangkauan. DHCP bisa memberikan informasi tentang IP yang disewakan, DNS, juga waktu penyewaan yang berlaku. Sehingga tidak ada lagi konfigurasi manual yang dilakukan.

Meskipun begitu, DHCP juga bisa menentukan bagian-bagian IP yang statis dan tidak akan berubah yang diberikan kepada host dengan MAC address tertentu.

DHCP bekerja dengan cara menawarkan diri dan IP yang disewakan, dengan melihat apakah masih ada alamat yang tersedia untuk disewakan. Lalu host meminta alamat yang disewakan tersebut lalu DHCP memberikanya informasi tentang IP, DNS, juga kapan penyewaan berakhir sehingga alamat bisa digunakan lagi.

Cara konfigurasi DHCP di Debian dengan menggunakan **dhcp3-server**, dengan sedikit konfigurasi yang menentukan jangkauan IP yang disewakan, DNS, gateway, broadcast, menentukan interface di mana dia bekerja. Konfigurasi ini bisa dilakukan dengan merubah file **/etc/dhcp/dhcpd.conf**. dan juga **/etc/default/dhcp**

Setelah konfigurasi, coba untuk melakukan koneksi dengan interface tempat DHCP server bekerja. Gunakan perintah untuk mengecek IP yang berlaku sesuai sistem operasi masing-masing. Apabila DHCP server sudah berjalan dengan lancar, maka host yang terkoneksi dengan DHCP server akan mendapatkan IP sesuai dengan range yang sudah ditentukan, juga DNS yang ditentukan, gateway, broadcast, netmask, dan informasi tentang batas penyewaan.

d. Tugas

1. Apakah yang dimaksud dengan DHCP?
2. Mengapa DHCP digunakan?
3. Bagaimana cara kerja DHCP?
4. Mengapa DHCP mempunyai batas waktu?
5. Apa kelebihan DHCP dibanding IP manual?
6. Apakah DHCP mungkin memberikan alamat IP yang sama pada dua host? Jelaskan!
7. Bagaimana cara DHCP mengatur agar IP tidak sama?
8. Apa nama DHCP server di Linux Debian?

9. Apabila subnet dengan IP router berbeda, apakah DHCP bisa berjalan? Jelaskan!
10. Apa saja yang diberikan oleh DHCP server?

e. Test Formatif

iii. DHCP merupakan kependekan dari ...
  a) Dynamic Host Content Provider
  b) Dynamic Host Control Provider
  c) Dynamic Host Control Protocol
  d) Dynamic Host Content Periode
  e) Dynamic Host Control Path

iv. DHCP digunakan supaya ...
  a) Pembagian konten ke pengguna tersebar
  b) Pembagian alamat web ke pengguna tersebar
  c) Pembagian alamat IP ke pengguna teratur
  d) Pembagian konten ke pengguna secara periodik
  e) Semua salah

v. DHCP cocok diterapan di ...
  a) Perusahaan besar
  b) Perusahaan kecil
  c) Kantor
  d) Semua benar
  e) Semua salah

vi. DHCP merupakan layanan dengan model komunikasi ...
  a) Client Server
  b) Peer to Peer
  c) SaaS
  d) Local Workstation
  e) Semua salah

vii. Apa yang terjadi apabila waktu kadaluarsa sudah tercapai?
    a) Pengguna meminta konten baru
    b) Pengguna meminta IP baru
    c) Server mempromosikan alamat tersebut tersedia
    d) Server mempromosikan kontent tersebut tersedia
    e) Semua salah

viii. Nama DHCP server di sistem operasi Linux Debian adalah ...
    a) DHCP-Server
    b) DHCP3-Server
    c) DHCP2-Server
    d) DHCP3
    e) DHCP2

ix. Tempat konfigurasi DHCP Debian ada di ...
    a) /etc/dhcp
    b) etc/dhcp
    c) /etc/conf/dhcp
    d) /bin/dhcp
    e) /conf/dhcp

x. Konfigurasi DHCP Server Debian ada di ...
    a) dhcpd.conf
    b) dhcpx.conf
    c) dhcpserver.conf
    d) dhclient.conf
    e) Semua salah

xi. Waktu kadaluarsa DHCP dihitung dalam satuan ...
    a) Mikrodetik
    b) Detik
    c) Menit
    d) Jam
    e) hari



xii. DHCP server tidak akan berfungsi apabila ...

a) Jumlah host terlalu banyak
b) Jumlah network terlalu banyak
c) Kapasitas jaringan rendah
d) Semua benar
e) Semua salah

f. Lembar Jawaban Tes Formatif

g. Lembar Kerja Siswa

18. Kegiatan Belajar 2 : Menyajikan Hasil Konfigurasi DHCP Server

a. Tujuan Pembelajaran

Setelah mengikuti kegiatan belajar 2 ini, siswa diharapkan dapat:

1. Menyajikan hasil konfigurasi DHCP server dan membuat laporan, makalah, dan presentasi.

b. Uraian Materi

xiii. Penyajian Hasil Konfigurasi DHCP Server.

Siswa ditugaskan untuk membuat kelompok yang terdiri dari 3 – 5 orang yang akan melakukan konfigurasi DHCP server dan diimplementasikan dalam jaringan kecil.

Siswa lalu diminta untuk membuat laporan tentang cara konfigurasi DHCP server, membuat tutorial, dan menuliskan kendala saat melakukan konfigurasi serta cara mengatasinya.

Siswa lalu ditugaskan untuk membuat makalah tentang implementasi DHCP server dalam kehidupan nyata, baik perusahaan atau kantor kecil. Lalu menyajikanya dalam bentuk presentasi yang kompak, singkat, dan padat.

c. Rangkuman

d. Tugas

1. Siswa ditugaskan untuk membuat kelompok yang terdiri dari 3 – 5 orang.
2. Siswa ditugaskan untuk melakukan konfigurasi DHCP server dengan 5 host.
3. Siswa ditugaskan untuk membuat tutorial konfigurasi.
4. Siswa ditugaskan untuk membuat laporan kendala yang dihadapi saat konfigurasi dan cara mengatasinya.
5. Siswa ditugaskan untuk membuat makalah tentang implementasi DHCP server dalam kehidupan nyata, baik dalam skala besar (perusahaan) atau kecil (kantor).
6. Siswa ditugaskan untuk menjadikan makalah tersebut kompak lalu menyajikanya dalam bentuk presentasi yang singkat, padat, dan jelas.

e. Tes Formatif

f. Lembar Jawaban Tes Formatif

g. Lembar Kerja Siswa

19. Kegiatan Belajar 1 : Memahami Cara Mengkonfigurasi DNS Server

a. Tujuan Pembelajaran

Setelah mengikuti kegiatan belajar 1 ini, siswa diharapkan dapat:

1. Memahami cara mengkonfigurasi DNS server.

b. Uraian Materi

1. **DNS Server**

1. Pengertian DNS Server



DNS adalah sebuah layanan untuk menyediakan penerjemahan alamat IP menjadi nama yang lebih mudah diingat. DNS menyediakan penerjemahan dua arah, dari IP menjadi nama, atau dari nama menjadi IP.

DNS berjalan di port 53, dia juga terhubung dengan layanan DNS lainya. DNS bekerja secara hirarki, di mana tingkatan-tingkatan DNS saling berkaitan dengan yang lainya.

DNS mempunyai database yang luas, berhati-hatilah memasukkan apa didalamnya, apabila DNS diisi dengan sampah, maka keluarnya juga akan sampah. Usahakan untuk mengatur DNS sekompak dan sekonsisten mungkin.

2. **Cara Kerja DNS Server**

DNS server menyimpan kamus dari IP menjadi nama atau sebaliknya. Pengguna melakukan permintaan terhadap DNS server dengan mengirimkan nama alamat yang dituju. DNS Server lalu mengembalikan kepada pengguna tersebut alamat IP yang dituju.

DNS server bekerja sama dengan server lainya, sehingga tidak terlalu membebani satu server. Oleh karena itu berbagai server DNS memiliki berbagai peran yang berbeda. Ada sebagian server yang hanya melakukan *forwarding*, yaitu meneruskan permintaan ke server lainya. Atau sebagian yang berfungsi sebagai *master* yang menyimpan data mapping IP ke nama domain dan sebaliknya.

3. **Optimasi DNS**

DNS tidak serta merta terus memberikan alamat, untuk membuat DNS bekerja secara optimal, pelanggan juga menyimpan hasil dari penerjemahan tersebut secara local. Penyimpanan sementara ini disebut dengan *caching*.

Ketika DNS server memberitahukan kepada pengguna bahwa www.aeonglobal.com mempunyai IP 54.55.12.20, maka pengguna yang bekerja sebagai DNS client, menyimpan sendiri di database kecilnya bahwa www.aeonglobal.com mempunyai IP 54.55.12.20, jadi ketika pengguna



mengakses www.aeonglobal.com, tidak perlu dilakukan permintaan kepada DNS server lagi.

4. **Jenis-jenis Name Server**

   Name server dikelompokan menjadi empat kelompok:

   1. Master

      Digunakan untuk meyimpan record-record zona original dan authoritative untuk namespace tertentu, menjawab pertanyaan dari name server lain yang berada dalam namespacenya.

   2. Slave

      Menjawab permintaan dari name server lain, dan hanya berfungsi sebagai cadangan dari server master.

   3. Caching Only

      Memberikan layanan resolusi yang sudah tersimpan, dan sama sekali tidak melakukan pengelolaan zona.

   4. Forwarding

      Melakukan penerusan terhadap permintaan ke name server lain.

5. **Name Server Debian**

   Pada sistem operasi Linux Debian, aplikasi untuk mengelola DNS adalah BIND, pada saat instalasi Debian, memilih untuk menginstal DNS Server akan membuat BIND otomatis ada di dalam sistem server.

   Apabila di instalasi server belum ada BIND, silahkan unduh dari ftp://ftp.isc.org/isc/bind9 .

   BIND menyimpan konfigurasi utamanya di **/etc/bind**.

```
root@serverone:/etc/bind# ls -l
total 52
-rw-r--r-- 1 root root 2544 Jul 27 17:10 bind.keys
-rw-r--r-- 1 root root  237 Jul 27 17:10 db.0
-rw-r--r-- 1 root root  271 Jul 27 17:10 db.127
-rw-r--r-- 1 root root  237 Jul 27 17:10 db.255
-rw-r--r-- 1 root root  353 Jul 27 17:10 db.empty
-rw-r--r-- 1 root root  270 Jul 27 17:10 db.local
-rw-r--r-- 1 root root 3048 Jul 27 17:10 db.root
-rw-r--r-- 1 root bind  463 Jul 27 17:10 named.conf
-rw-r--r-- 1 root bind  490 Jul 27 17:10 named.conf.default-zones
-rw-r--r-- 1 root bind  165 Jul 27 17:10 named.conf.local
-rw-r--r-- 1 root bind  572 Jul 27 17:10 named.conf.options
-rw-r----- 1 bind bind   77 Dec 13 10:40 rndc.key
-rw-r--r-- 1 root root 1317 Jul 27 17:10 zones.rfc1918
root@serverone:/etc/bind# _
```

Konfiguras utama dari BIND ada di file **/etc/bind/named.conf**.

Karena DNS, nama domain di atur oleh organisasi internasional (IANA), maka DNS kita juga harus bekerja sama dengan name server dunia. Definisi name server dunia ada di **/etc/bind/db.root**.

Sebelum melanjutkan lebih lagi, Linux menaruh file tentang informasi name server sebagai client di **/etc/resolv.conf**.

Coba baca isi dari file **/etc/resolv.conf**, gunakan **cat /etc/resolv.conf**.

```
search portal.aeonglobal.com aeonglobal.com
nameserver 5.45.75.11
nameserver 95.211.156.101

                              [ Wrote 3 lines ]

root@serverone:/etc/bind# _
```

Baris pertama, mengindikasikan dimana harus pertama kali mencari dari domain yang diminta. Misalkan, kita mencoba mengakses **ion**, maka yang pertama di coba adalah **ion.portal.aeonglobal.com**, lalu **ion.aenglobal.com**

lalu akhirnya mencoba mencari menggunakan nameserver yang ada di bawah.

Sebisa mungkin jangan menaruh domain di baris search, karena akan memakan waktu ketika mencarinya.

Karena kita menjadikan mesin server kita sendiri sebagai name server, maka ganti nameserver menjadi,

```
nameserver 127.0.0.1
```

Sehingga apabila ada permintaan, yang digunakan sebagai server DNS adalah server kita sendiri.

Sekarang, coba buka kembali direktori **/etc/bind**, gunakan **cd** supaya tidak berkali-kali memberikan alamat absolut.

Ada sebuah file **named.conf**, gunakan **nano** sebagai text editor untuk mengedit isi dari **named.conf**.

```
// This is the primary configuration file for the BIND DNS server named.
//
// Please read /usr/share/doc/bind9/README.Debian.gz for information on the
// structure of BIND configuration files in Debian, *BEFORE* you customize
// this configuration file.
//
// If you are just adding zones, please do that in /etc/bind/named.conf.local

include "/etc/bind/named.conf.options";
include "/etc/bind/named.conf.local";
include "/etc/bind/named.conf.default-zones";


zone "aeonglobal.com" {
 type master;
 file "/etc/bind/aeonglobal.com";
};



                              [ Wrote 17 lines ]

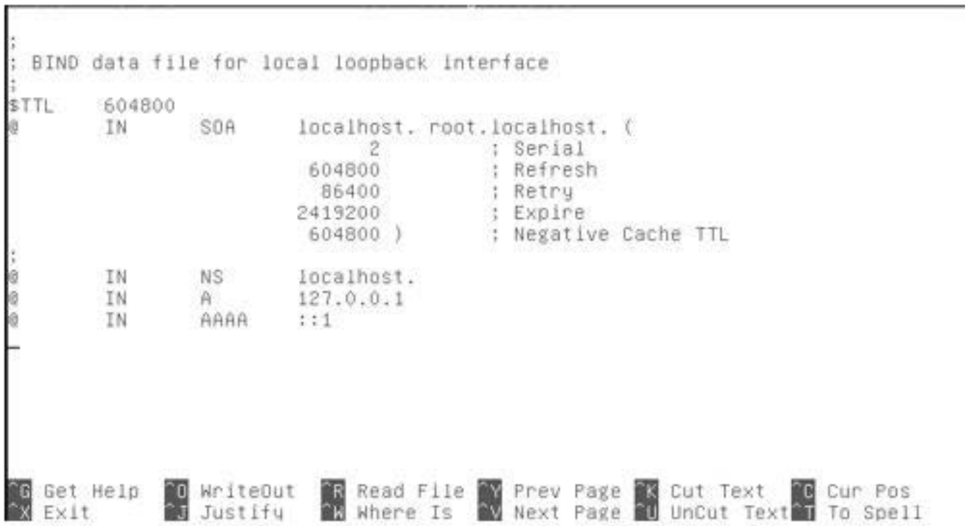
root@serverone:/etc/bind# _
```

Rubah isi dari named.conf sehingga seperti di atas. Kita membuat sebuah *mapping* bahwa domain **aeonglobal.com** akan diproses dengan aturan yang ada di file **/etc/bind/aeonglobal.com**.

Setelah itu, coba buat file bernama **aeonglobal.com** di direktori **/etc/bind**, gunakan

```
cp db.local aeonglobal.com
```

Untuk membuat file **aenglobal.com** berdasarkan file **db.local** sebagai template.



```
;
; BIND data file for local loopback interface
;
$TTL    604800
@       IN      SOA     localhost. root.localhost. (
                              2         ; Serial
                         604800         ; Refresh
                          86400         ; Retry
                        2419200         ; Expire
                         604800 )       ; Negative Cache TTL
;
@       IN      NS      localhost.
@       IN      A       127.0.0.1
@       IN      AAAA    ::1

^G Get Help   ^O WriteOut   ^R Read File  ^Y Prev Page  ^K Cut Text   ^C Cur Pos
^X Exit       ^J Justify    ^W Where Is   ^V Next Page  ^U UnCut Text ^T To Spell
```

Gunakan nano untuk membuka isi dari aeonglobal.com, dan diatas inilah isinya.

Rubahlah file di atas menjadi seperti ini,



```
GNU nano 2.2.4                File: aeonglobal.com

;
; BIND data file for local loopback interface
;
$TTL    604800
@       IN      SOA     aeonglobal.com. root.aeonglobal.com. (
                              2         ; Serial
                         604800         ; Refresh
                          86400         ; Retry
                        2419200         ; Expire
                         604800 )       ; Negative Cache TTL
;
@       IN      NS      aeonglobal.com.
@       IN      A       127.0.0.1

                        [ Read 13 lines ]
root@serverone:/etc/bind# _
```

Kita membuat konfigurasi bahwa alamat aeonglobal.com itu mempunyai IP 127.0.0.1.

Setelah perubahan selesai, lakukan restart pada BIND dengan menggunakan **init.d**,

```
/etc/init.d/bind9 restart
```

Tunggu beberapa saat hingga BIND selesai merestart. Setelah itu, gunakan perintah **dig** untuk mengetahui informasi tentang domain yang baru saja kita buat.

```
dig aeonglobal.com
```

```
root@serverone:~# dig aeonglobal.com

; <<>> DiG 9.7.3 <<>> aeonglobal.com
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 60616
;; flags: qr aa rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 1, ADDITIONAL: 0

;; QUESTION SECTION:
;aeonglobal.com.                    IN      A

;; ANSWER SECTION:
aeonglobal.com.         604800  IN      A       127.0.0.1

;; AUTHORITY SECTION:
aeonglobal.com.         604800  IN      NS      aeonglobal.com.

;; Query time: 50 msec
;; SERVER: 127.0.0.1#53(127.0.0.1)
;; WHEN: Sat Dec 14 19:40:37 2013
;; MSG SIZE  rcvd: 62

root@serverone:~# _
```

Dengan menggunakan **dig**, kita bisa mengetahui detail dari query DNS yang kita lakukan. Bahwa domain aeonglobal.com mempunyai IP 127.0.0.1. Waktu untuk memprosesnya 50 mili second.

Ingat bahwa mapping DNS tidak hanya dari nama menjadi IP, tapi juga sebaliknya. Oleh karena itu kita juga akan melakukan konfigurasi untuk mebuat konfigurasi mapping dari IP menjadi domain.

Buka kembali file **named.conf**, lalu tambahkan baris di bawah sehingga jadi seperti ini.

```
  GNU nano 2.2.4              File: named.conf


zone "aeonglobal.com" {
 type master;
 file "/etc/bind/aeonglobal.com";
};

zone "1.168.192.in-addr.arpa" {
 type master;
 file "/etc/bind/aeonglobal.com.rev";
};
_

                              [ Read 22 lines ]
^G Get Help   ^O WriteOut   ^R Read File  ^Y Prev Page  ^K Cut Text   ^C Cur Pos
^X Exit       ^J Justify    ^W Where Is   ^V Next Page  ^U UnCut Text ^T To Spell
```

Yang perlu dimasukkan hanyalah IP yang mengidentifikasikan networknya, lalu diteruskan dengan in-addr.arpa. ARPA adalah instansi yang bertanggung

jawab atas internet. Proyek internet pertama kali dikembangkan untuk militer, dengan nama ARPANET.

Setelah itu, tentukan tipenya sebagai master, lalu penentuan file definisi dengan nama **aeonglobal.com.rev**. Karena belum ada file bernama **aeonglobal.com.rev**, maka kita akan membuatnya dengan cara menyalin template dari **db.127**.

```
cp db.127 aeonglobal.com.rev
```

Setelah itu gunakan nano dan edit file **aeonglobal.com.rev** yang semula seperti ini,

```
  GNU nano 2.2.4              File: aeonglobal.com.rev

;
; BIND reverse data file for local loopback interface
;
$TTL    604800
@       IN      SOA     localhost. root.localhost. (
                              1         ; Serial
                         604800         ; Refresh
                          86400         ; Retry
                        2419200         ; Expire
                         604800 )       ; Negative Cache TTL
;
@       IN      NS      localhost.
1.0.0   IN      PTR     localhost.



                              [ Read 13 lines ]
^G Get Help  ^O WriteOut  ^R Read File ^Y Prev Page ^K Cut Text  ^C Cur Pos
^X Exit      ^J Justify   ^W Where Is  ^V Next Page ^U UnCut Text^T To Spell
```

menjadi seperti ini.

```
  GNU nano 2.2.4              File: aeonglobal.com.rev

;
; BIND reverse data file for local loopback interface
;
$TTL    604800
@       IN      SOA     aeonglobal.com. root.aeonglobal.com. (
                              1         ; Serial
                         604800         ; Refresh
                          86400         ; Retry
                        2419200         ; Expire
                         604800 )       ; Negative Cache TTL
;
@       IN      NS      aeonglobal.com.
1       IN      PTR     aeonglobal.com.



                              [ Read 13 lines ]
^G Get Help  ^O WriteOut  ^R Read File ^Y Prev Page ^K Cut Text  ^C Cur Pos
^X Exit      ^J Justify   ^W Where Is  ^V Next Page ^U UnCut Text^T To Spell
```

Perlu perhatikan di bagian baris terakhir, ada angka **1** di depan. Angka itu menunjukan IP host dari yang sudah kita konfigurasi di **named.conf**. Di **named.conf** kita menunjukkan bahwa **aeonglobal.com** mempunyai IP 1.168.192 atau secara terbalik 192.168.1. Karena di bagian **aeonglobal.com.rev** kita definisikan angka 1, berarti IP lengkap dari **aeonglobal.com** adalah 192.168.1.1.

Sekarang coba gunakan **nslookup** untuk mengetahui nama dari IP.

```
nslookup <nama_domain_atau_ip>
```

Karena kita ingin tahu apa nama domain dari 192.168.1.1, gunakan

```
nslookup 192.168.1.1
```

Jawabanya adalah 192.168.1.1 mempunyai nama **aeonglobal.com**. Meskipun **aeonglobal.com** mengarah ke 127.0.0.1, tapi definisi *reverse-lookup* yaitu pencarian nama dari IP dengan pencarian IP dari nama bersifat independen.

Mengatur Sub Domain

Apabila anda perhatikan, ada sebuah situs yang mempunyai sub domain. Sub domain adalah domain tambahan yang ada di domain utama. Jadi misalkan kita mempunyai sebuah alamat **aeonglobal.com** kita bisa memberinya sub domain sehingga berbentuk **research.aeonglobal.com**.

Kita coba membuat sebuah domain baru bernama **store.com** (dengan IP 192.168.2.1). Sebuah alamat website dari situs yang menjual berbagai macam barang. Untuk tiap kategori dari barang yang dijual kita akan membuatkan sub domainya. Kali ini kita cukup membuat 2 sub domain, yaitu **apps** dan **webs**. Sehingga nanti ada 3 alamat, yaitu **store.com**, **apps.store.com**, dan **webs.store.com**.



```
 GNU nano 2.2.4                File: named.conf


:one "store.com" {
 type master;
 file "/etc/bind/store.com";
:;

:one "2.168.192.in-addr.arpa" {
 type master;
 file "/etc/bind/store.com.rev";
:;




^G Get Help  ^O WriteOut  ^R Read File ^Y Prev Page ^K Cut Text  ^C Cur Pos
^X Exit      ^J Justify   ^W Where Is  ^V Next Page ^U UnCut Text^T To Spell
```

Rubah **named.conf** dengan menambahkan baris sehingga menjadi seperti di atas, kita akan membuat file lagi dengan nama **store.com** dan **store.com.rev** sama seperti langkah sebelumnya, dengan menyalin **db.local** dan **db.127** lalu merubahnya.

/etc/bind/store.com

```
 GNU nano 2.2.4                File: store.com

;
; BIND data file for local loopback interface
;
$TTL    604800
@       IN      SOA     store.com. root.store.com. (
                              2         ; Serial
                         604800         ; Refresh
                          86400         ; Retry
                        2419200         ; Expire
                         604800 )       ; Negative Cache TTL
;
@       IN      NS      store.com.
@       IN      A       192.168.2.1
apps    IN      CNAME   @
webs    IN      CNAME   @



                              [ Wrote 15 lines ]

root@serverone:/etc/bind# _
```

/etc/bind/store.com.rev

```
 GNU nano 2.2.4              File: store.com.rev

;
; BIND reverse data file for local loopback interface
;
$TTL    604800
@       IN      SOA     store.com. root.store.com. (
                              1         ; Serial
                         604800         ; Refresh
                          86400         ; Retry
                        2419200         ; Expire
                         604800 )       ; Negative Cache TTL
;
@       IN      NS      store.com.
1       IN      PTR     apps.store.com.
1       IN      PTR     webs.store.com.



                              [ Wrote 14 lines ]

root@serverone:/etc/bind#
```

Setelah selesai dengan semua file-file yang dibutuhkan, jangan lupa untuk melakukan restart pada bind. Lalu gunakan **nslookup** untuk melakukan uji coba.

```
nslookup store.com
nslookup apps.store.com
nslookup webs.store.com
```

Hasil dari nslookup akan terlihat seperti ini.

```
root@serverone:/etc/bind# nslookup store.com
Server:         127.0.0.1
Address:        127.0.0.1#53

Name:   store.com
Address: 192.168.2.1

root@serverone:/etc/bind# nslookup apps.store.com
Server:         127.0.0.1
Address:        127.0.0.1#53

apps.store.com  canonical name = store.com.
Name:   store.com
Address: 192.168.2.1

root@serverone:/etc/bind# nslookup webs.store.com
Server:         127.0.0.1
Address:        127.0.0.1#53

webs.store.com  canonical name = store.com.
Name:   store.com
Address: 192.168.2.1

root@serverone:/etc/bind# _
```

Lalu cobalah untuk melakukan reverse-lookup dengan cara memasukkan IP dari **store.com**.

```
root@serverone:/etc/bind# nslookup 192.168.2.1
Server:         127.0.0.1
Address:        127.0.0.1#53

1.2.168.192.in-addr.arpa        name = apps.store.com.
1.2.168.192.in-addr.arpa        name = webs.store.com.

root@serverone:/etc/bind# _
```

Rangkuman

DNS merupakan sebuah layanan yang merubah permintaan dari alamat IP menjadi nama domain atau dari nama domain menjadi nama IP. DNS muncul karena tidak semua orang bisa menghafal deretan angka seperti 152.52.34.1. Karena itulah DNS digunakan sehingga alamat 152.52.34.1 bisa berubah menjadi alamat.com yang lebih mudah diingat.

DNS server bekerja sama satu sama lain, karena itu DNS server mempunyai 4 macam fungsi, yaitu sebagai master untuk menyimpan data *mapping* (terjemahan dari IP ke nama domain dan sebaliknya), lalu slave, sebagai cadangan dari tipe master, lalu forward, sebagai penerus permintaan, lalu sebagai caching untuk melakukan penyimpanan sementara terhadap permintaan yang sudah pernah dilakukan.



DNS bekerja di atas port 53, dan melayani permintaan dari pengguna yang juga bekerja dengan protocol DNS. Pada sistem operasi Linux Debian, cara untuk mengatur DNS bisa dilakukan dengan menggunakan BIND.

BIND merupakan penyedia DNS yang tergabung dengan paket Linux Debian (apabila saa instalasi dicentang DNS server), atau bisa diunduh dengan menggunakan manajer paket dari Debian. Yaitu dengan menggunakan **apt-get install bind9**.

BIND menyimpan konfigurasi utama di direktori **/etc/bind**. Ada beberapa file yang berhubungan dengan DNS di sana. File **/etc/bind/db.root** menyimpan informasi tentang DNS di seluruh dunia yang harus diatur dan diupdate secara berkala. File /**etc/bind/named.conf** merupakan konfigurasi DNS utama di mana kita bisa menambahkan deskripsi mapping dan informasi lainya.

Sebuah domain bisa mempunyai sub domain. Sub domain ini digunakan untuk membuat *SoC (Separation of Concers)* atau pemecahan konsentrasi sehingga sebuah sistem teratur dan rapi.

c. Tugas

1. Apa yang dimaksud dengan DNS?
2. Bagaimana DNS merubah IP menjadi domain dan sebaliknya?
3. Sebutkan dan jenis-jenis server di DNS!
4. Sebutkan dan jelaskan langkah-langkah installasi BIND!
5. Buatlah dan jelaskan langkah-langkah untuk konfigurasi mapping untuk domain www.afoundation.org ke IP 192.168.1.10!
6. Mengapa DNS digunakan?
7. Apakah ada cara lain untuk membuat pengguna lebih mudah mengakses sebuah IP selain DNS?
8. Apa kegunaan dari perintah DIG?
9. Apa yang dimaksud dengan reverse-lookup?
10. Apa perbedaan dari reverse-lookup dengan lookup?



d. Test Formatif

1. Apa saja yang bisa dilakukan oleh DNS?
   a) Merubah IP ke domain
   b) Merubah domain ke IP
   c) Merubah domain ke Mac
   d) Jawaban A dan C benar
   e) Jawaban A dan B benar
2. DNS berjalan di port ...
   a) 11
   b) 35
   c) 43
   d) 53
   e) 64
3. DNS menggunakan optimasi untuk meminimalisir request, disebut dengan ...
   a) Request Optimizer
   b) Cache
   c) Optimizer
   d) DNS Request Halt
   e) Halt
4. Yang bukan termasuk jenis name server adalah ...
   a) Master
   b) Slave
   c) Dispatcher
   d) Caching
   e) Forwarding
5. Aplikasi untuk mengelola DNS di Debian adalah ...
   a) DINB
   b) BIN
   c) BIND
   d) DNSCloud
   e) DNSDaddy



6. File berikut adalah file konfigurasi BIND, kecuali ...
    a) named.conf
    b) db.local
    c) db.root
    d) db.conf
    e) local.conf
7. Linux Debian menyimpan file konfigurasi DNS client di ...
    a) /etc/resolve.conf
    b) /etc/resolves.conf
    c) /etc/resolv.conf
    d) etc/resolv.conf
    e) /resolc.fconf
8. Perintah yang digunakan untuk analisa lookup DNS adalah ...
    a) DGI
    b) DIG
    c) GID
    d) GDI
    e) IDG
9. DNS bekerja secara hierarkial, yang berarti ...
    a) Satu lain saling terhubung
    b) Mempunyai jabatan dan susunan yang teratur
    c) Terpisah satu sama lain
    d) Berinteraksi dengan satu sama lain
    e) Terpusat dengan satu server
10. DNS menggunakan metode caching, yang berarti ...
    a) Alamat yang sudah pernah di cari tidak akan dicari lagi dalam jangka waktu tertentu
    b) Alamat yang sudah di cari akan dicari lagi
    c) Alamat yang belum ada akan terus dicari tanpa menghasilkan error
    d) Alamat yang tidak ada akan menghasilkan error
    e) Semua alamat harus ada dan tidak boleh ada error

e. Lembar Jawaban Tes Formatif

f. Lembar Kerja Siswa

20. Kegiatan Belajar 2 : Menyajikan Hasil Konfigurasi DNS Server

Tujuan pembelajaran

Setelah mengikuti kegiatan belajar 2 ini, siswa diharapkan dapat:

1. Menyajikan hasil dari konfigurasi DNS Server pada sistem operasi Linux dalam bentuk praktek, laporan, makalah, dan presentasi.

a. Uraian Materi

2. **Menyajikan hasil konfigurasi DNS server.**

Siswa ditugaskan untuk melakukan konfigurasi DNS server yang lengkap, mulai dari lookup sampai reverse lookup berkelompok dengan anggota 3 – 5 orang.

Siswa lalu ditugaskan untuk membuat laporan tentang proses konfigurasi DNS, kendala yang dihadapi, dan cara menyelesaikanya. Serta mencatat batas kemampuan DNS beserta kegunaanya.

Siswa lalu membuat makalah tentang struktur DNS di dunia, dibagi-bagi berdasarkan benua, atau negara. Setelah itu melakukan presentasi dengan makalah tersebut yang sudah dikompakkan dan dibuat singkat, padat, dan cepat.

b. Rangkuman

c. Tugas

3. Siswa membuat kelompok yang terdiri dari 3 – 5 orang.
4. Siswa membuat makalah tentang proses konfigurasi DNS, kendala yang dihadapi, penyelesaian, dan mencatat batas kemampuan DNS.

5. Siswa membuat makalah tentang struktur DNS di dunia berdasarkan benua, atau negara.
6. Siswa melakukan presentasi berdasarkan makalah yang sudah diolah untuk dijadik presentasi yang padat, jelas, dan singkat.

d. Tes Formatif

e. Lembar Jawaban Tes Formatif

f. Lembar Kerja Siswa

21. Kegiatan Belajar 1 : Memahami Cara Mengkonfigurasi Web atau HTTP Server

Tujuan Pembelajaran

Setelah mengikuti kegiatan belajar 1 ini, siswa diharapkan dapat :

xiv. Memahami cara mengkonfigurasi web ata HTTP server.

Uraian Materi

xv. Web Server

1. Pengertian Web Server

Salah satu layanan yg paling sering disediakan oleh server adalah layanan web. Salah satu alasanya dari penggunaan web server adalah dia bebas platform. Semua aplikasi yang berjalan di atas web hanya perlu menyesuaikan dengan web browser pengguna.

Web server adalah sebuah penyedia layanan dengan protocol HTTP. Web server juga dikenal dengan nama HTTP server, menyediakan kemampuan untuk mengirimkan dokumen *hyper-text* kepada pengguna. Dokumen hypertext itu nantinya digunakan untuk dijadikan tampilan.

Web server normal, berjalan di atas port 80. Versi aman dari web server, atau HTTPS server, berjalan di atas port 443. HTTP server biasa, hanya menyediakan akses ke file-file yang berada di dalam folder server HTTP.

HTTPS dilindungi dengan enkripsi sehingga data yang terkirim dijamin keamananya. Situs-situs web yang membutuhkan keamanan ekstra seperti



bank, transaksi, email, penyimpanan file menggunakan HTTPS. Meskipun beberapa tidak menggunakan HTTPS, tapi mau tidak mau web-web kritikal seperti itu harus menggunakan HTTPS demi mendapatkan pelanggan.

Ketika HTTP server melayani pengguna, kebanyakan data yang dikirm berformat HTML. Tapi HTML sendiri bekerja kompak dengan file dari format lainya, seperti CSS untuk mengatur tampilan, JS untuk mengatur interaksi, dan tambahan-tambahan lainya seperti gambar.

Biasanya, ketika pengguna melakukan permintaan terhadap HTTP Server, HTTP server membalas dengan menyediakan berkas asli yang ada di server. Tapi, tidak harus seperti itu, sesuai dengan bagaimana HTTP server itu diimplementasikan.

HTTP server yang dinamis, apabila ingin menambahkan fungsi-fungsi di sistemnya, bisa melalui dua cara. Yaitu dengan pembuatan ulang software HTTP server, atau menggunakan bahasa pemrograman di sisi server atau *server side scripting*. Dengan menggunakan bahasa seperti ASP, atau PHP, maka HTTP server bisa menyediakan fungsi yang berbeda-beda tanpa perlu menulis ulang kode untuk software HTTP server itu sendiri.

HTTP server tidak serta merta hanya menyediakan data kepada pengguna, HTTP server juga bisa meminta data dari pengguna baik menggunakan form, ataupun dengan pengunggahan file.

HTTP server sekarang tidak hanya untuk menyediakan layanan melalui web server yang berbentuk mesin besar. Sekarang, beberapa alat menggunakan HTTP sebagai antar muka dengan pengguna, seperti router, printer, webcam. Tapi di alat-alat kecil ini biasanya HTTP digunakan dalam jaringan lokal saja.

2. **Jenis Server HTTP**

Ada dua jenis server HTTP, yaitu Kernel dan User. Versi kernel dari HTTP server mempunyai kecepatan yang lebih karena langsung terintegrasi dengan OS dan memang ditujukan untuk HTTP server. Sehingga

mempunyai akses langsung ke sumber daya tingkat bawah seperti adaptor jaringan, buffer, atau memory.

Jenis user lebih lama dan terbatas, karena mereka tidak mempunyai akses langsung ke dalam hardware dan sumber daya tingkat bawah. Mereka harus meminta dahulu kepada OS untuk sumber daya, lalu mereka harus berbagi resource dengan aplikasi lainya.

Pada sistem Windows, bisa menggunakan IIS, atau di Linux menggunakan TUX.

3. **Batas-Batas Web Server**

Web server terbatas, dia mempunyai keterbatasan tentang berapa jumlah pengguna yang bisa dilayani secara bersamaan. Karena itu biasanya ada lebihd ari satu web server yang disediakan apabila jumlah pengguna yang dilayani sangat besar.

Batas-batas web server ini ditentukan bebera faktor.

- Konfigurasi Web Server
- Jenis HTTP Request
- Jenis Konten Dinamik atau Statik
- Batas Hardware atau Software

Ketika batas ini tercapai, maka web server akan mengalami macet dan tidak responsif.

Biasanya, batas-batas ini tercapai ketika ada beberapa hal yang terjadi.

- Banyaknya pengguna yang terhubung secara bersamaan.

  Ketika ada pengguna yang terhubung secara bersamaan. Maka sumber daya dari web server akan terbagi-bagi juga dalam saat yang bersamaan. Proses input output memakan waktu yang lama, pembacaan file yang berkali-kali dalam waktu yang tidak jauh berbeda akan membuat web server kehilangan keseimbangan dan akhirnya tidak bisa menyelesaikan permintaan.
- Serangan DDOS



Serangan DDOS atau Distributed Denial Of Service, membuat server kewalahan melayani serangan tersebut. Akibatnya, pengguna lainya tidak akan mendapatkan sumber daya yang diminta.

- Virus

  Adanya virus yang menyebar melalui jutaan komputer bisa menghambat kinerja dari server. Apabila ada satu juta komputer terinfeksi, maka mereka akan mencoba menyebar melewati jaringan internet, dan salah satu layanan yang juga terkena serangan adalah web server.
- Jaringan Lambat

  Jaringan yang lambat akan mengakibatkan server memrposes suatu permintaan lebih lama daripada biasanya. Hal yang seperti ini menyebabkan *bottleneck*, jadi ketika seharusnya server selesai memproses, dia harus menunggu lama karena jaringanya lambat, hinga akhirnya semua beban menumpuk dan server akhirnya down.

Beberapa cara juga bisa dilakukan untuk membuat batas-batas web server tidak tercapai. Dengan mengoptimalkan sumber daya yang ada, dan kemampuan dari web server.

- Mengatur Trafik Jaringan

  Mengatur trafik jaringan dengan membatasi sebuah permintaan dengan firewall, dan juga memfilter permintaan HTTP yang buruk, juga melakukan pengaturan bandwidth.
- Menggunakan Cache

  Web server harus menggunakan teknologi cache, sehingga resource yang sama diminta berkali-kali tidak perlu melewati proses yang lama. Apalagi melewati jalur IO yang padat dan lambat.
- Menggunakan Lebih dari Satu Web Server

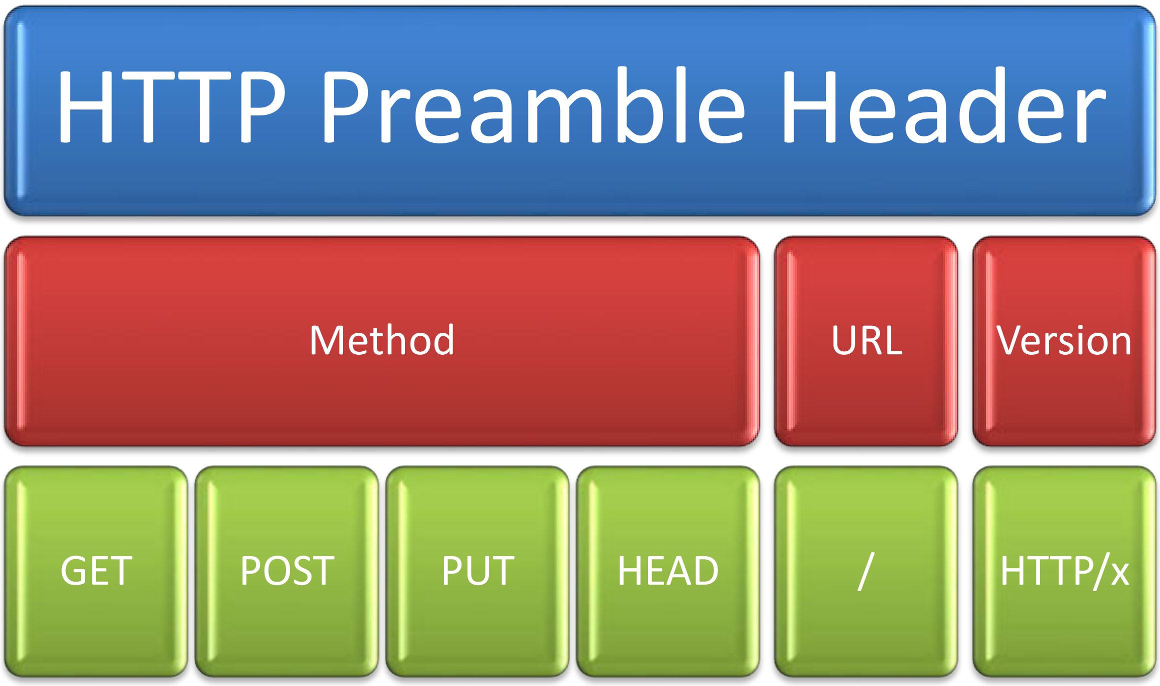
  Menggunakan satu web server akan membuat sistem terbebani, namun apabila menggunaan dua web server, maka server satunya bisa beristirahat dan membebaskan sumber daya yang sudah diproses, dan web server lainya mengerjakan beberapa dari tanggung jawab web server satu.



Memecah web server ini merupakan salah satu cara yang sangat efektif, namun karena faktor biaya membuat lebih dari web server bukanlah harga yang murah.

4. **Protocol HTTP**

Setiap permintaan HTTP mempunyai dua bagian, yaitu header dan content. Header untuk permintaan dan header untuk pelayanan berbeda. HTTP server hanya merespon apabila ada permintaan. Permintaan HTTP diberikan



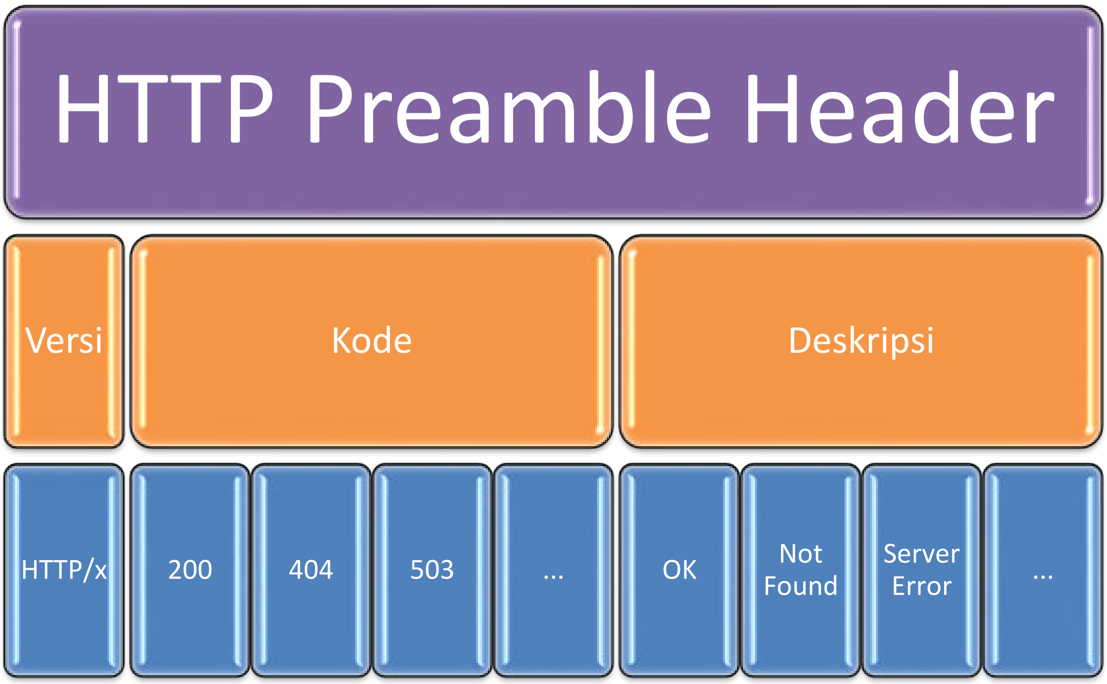
melalui URL, juga metode HTTP yang digunakan.

Header pembuka HTTP, terdiri dari tiga bagian. Metode, URL, dan versi HTTP. Method bisa berisi GET, POST, PUT, HEAD. URL bisa berisi alamat sumber daya yang diminta, dan versi mengindikasikan versi protocol HTTP (sekarang 1.1 saat ini ditulis)

HTTP server melakukan respon yang berbeda-beda seiring dengan perbedaan metode, url atau versi.

Metode GET, digunakan untuk mengambil data. Sedangkan metode POST, digunakan untuk memasang data. Metode PUT, digunakan untuk menaruh data, dan metode HEAD digunakan untuk meminta informasi.

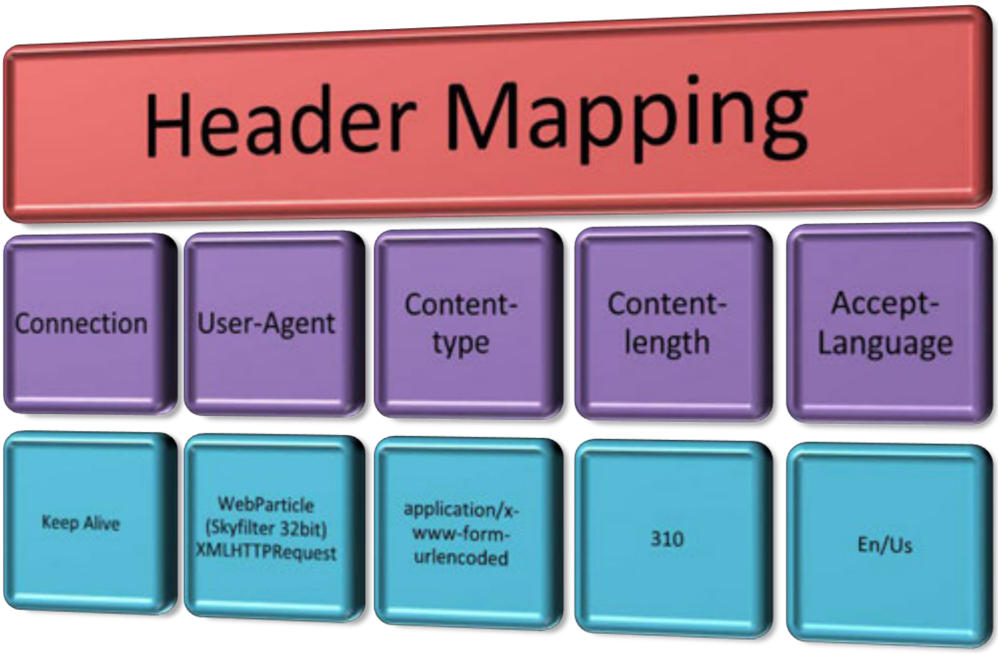
Berbeda dengan header pembuka permintaan, header pembuka respon



bentuknya seperti ini.

Setelah header pembuka, akan ada lagi header-header lainya. Header ini berbentuk mapping, seperti kamus, dengan nama dan niainya.

Header berbentuk mapping ini jumlahnya lebih banyak. Tidak mengikuti pola tertentu, dan diakhiri dengan baris kosong.



Gambar di atas merupakan struktur dari header mapping dari sisi permintaan. Isinya tidak terbatas, bahkan kita bisa menambahkan header sesuai dengan kebutuhan kita.

Fomat dari header mapping, mempunyai format **<nama_mapping> : <nilai_mapping>**, seperti **Connection: Keep-Alive**

Content-Type

text/html

Content-Length

10849

Host

aeonglobal.com

Encoding

UTF-8

Tipikal header respon dari server. Sama, dia juga tidak terbatas dan bisa disesuaikan dengan kebutuhan.

Contohnya, apabila di web server ada sebuah file dengan nama **index.html**¸maka cara untuk mendapatkan file tersebut. Pengguna harus mengirimkan kepada web server permintaan yang berbentuk seperti ini.

```
GET /index.html HTTP/1.1
Connection: Keep-Alive
User-Agent: WebParticle (Skyfilter 32bit) XMLHTTPRequest
```

Dengan mengirimkan permintaan seperti di atas, maka server akan merespon dengan bentuk seperti ini.

```
HTTP/1.1 200 OK
Content-type: text/html
Content-length: 520
```




```
<!DOCTYPE HTML>
<html>
    …
    …
    …
</html>
```

Setelah header mapping terakhir, ada satu baris kosong. Di bagian header mapping permintaan juga ada baris kosong, namun tidak terlihat karena pengguna yang meminta layanan tidak memberikan data apa-apa.

5. **Jenis HTTP Server**

   Tidak semua HTTP server bekerja dengan cara yang sama. Ada HTTP server yang hanya menyediakan file statis, ada juga yang menyediakan file dinamis. Selama aplikasi tersebut melayani permintaan sesuai dengan aturan HTTP. Maka dia disebut dengan HTTP Server.

   Staticlet

   Staticlet adalah web server yang tidak begitu populer, kemampuanya terbatas dan hanya bisa melayani pemberian file statis. Tidak bisa dikonfigurasi, dan lebih berguna sebagai portal untuk membuat file sharing satu arah berbasis HTTP.

   Apache

   Apache merupakan salah satu web server yang populer, dilengkapi dengan kemampuan bahasa PHP, CGI (Perl, Python), juga konfigurasi menggunakan file .ht. Apache bisa melakukan tugasnya dengan baik, konfigurasinya juga banyak dan sangat mudah dipelajari.

   Nginx

   Nginx adalah salah satu web server yang bekerja dengan cepat. Nginx menggunakan C++, salah satu bahasa yang bekerja dengan cepat.

   IIS



IIS adalah web server dari Microsoft, kemampunya dalah dia mampu menggunakan dan mengoptimalkan teknologi Microsoft seperti .NET, ASP, WCF, Hyper-V, dll.

GlassFish

GlassFish adalah salah satu server yang menggunakan Java sebagai platformnya. Java sendiri merupakan platform independen yang bisa berjalan di berbagai paltform.

Apache terinstall bersamaan saat kita menginstall Debian apabila kita mencentang Web Server pada saat installasi.

6. **Apache di Debian**

   Apabila saat menginstal Debian dicentang opsi untuk menginstal Web Server, maka Apache sudah akan terinstal. Apabila belum, gunakan **apt** untuk menginstalnya.

```
apt-get install apache2
```

Untuk melihat apakah Apache sudah ada di server, coba lihat apakah ada direktori **/etc/apache2**. Lalu, gunakan **wget** untuk mencoba Apache.

```
wget localhost –O /usr/index.html
```

Apabila tampilan yang didapat seperti ini, maka Apache sudah berjalan dengan lancar.

```
root@serverone:~# wget localhost -O /usr/index.html
--2013-12-15 11:20:02--  http://localhost/
Resolving localhost... 127.0.0.1
Connecting to localhost|127.0.0.1|:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 177 [text/html]
Saving to: "/usr/index.html"

100%[======================================>] 177         --.-K/s   in 0s

2013-12-15 11:20:02 (4.62 MB/s) - "/usr/index.html" saved [177/177]

root@serverone:~# _
```

Berarti Apache yg sekarang berjalan di server kita (localhost), sudah berjalan. Untuk melihat halaman web yang barusan kita unduh, kita bisa menggunakan perintah **cat.**

```
cat /usr/index.html
```

Hasilnya seperti ini,

```
root@serverone:~# cat /usr/index.html
<html><body><h1>It works!</h1>
<p>This is the default web page for this server.</p>
<p>The web server software is running but no content has been added, yet.</p>
</body></html>
root@serverone:~# _
```

Isi dari index.html yang baru saja kita buka adalah seperti di atas. Apabila ingin melakukan perubahan terhadap file **index.html**, Apache menyimpan file yang digunakan oleh server di direktori **/var/www**.





Hanya ada file **index.html**, sekarang coba rubah isi dari file **index.html** menjadi seperti di bawah ini untuk membuktikan bahwa itu benar-benar file yang disediakan oleh Apache.



Lalu coba gunakan **wget** lagi untuk mendapatkan isi dari file yang disediakan localhost. Apabila isi dari file yang didapatkan oleh **wget** berisi seperti di atas. Berarti sudah benar bahwa file **index.html** itulah yang disediakan oleh Apache di server kita.

Mengingat saat konfigurasi DNS kita sudah pernah membuat domain dengan nama **aeonglobal.com** yang mengarah ke server kita sendiri, coba

gunakan **wget** untuk melihat apakah **aeonglobal.com** sama dengan **localhost**.

```
wget aeonglobal.com –O /usr/index2.html
```

```
root@serverone:/etc/apache2# wget aeonglobal.com -O /usr/index2.html
--2013-12-15 15:50:05--  http://aeonglobal.com/
Resolving aeonglobal.com... 127.0.0.1
Connecting to aeonglobal.com|127.0.0.1|:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 23 [text/html]
Saving to: "/usr/index2.html"

100%[======================================>] 23          --.-K/s   in 0s

2013-12-15 15:50:05 (388 KB/s) - "/usr/index2.html" saved [23/23]

root@serverone:/etc/apache2# _
```

Pasti isinya sama karena **aeonglobal.com** merujuk sama seperti **localhost** merujuk, yaitu ke 127.0.0.1.

Karena Apache bisa dikonfigurasi, kita akan mencoba untuk melakukan perubahan konfigurasi pada Apache. Masuk ke folder utama Apache yang berada di **/etc/apache2**.

Kita akan menambahkan subdomain di **aeonglobal.com** sehingga dia mempunyai sub domain **hello** dan **bye**.

Rubah file **/etc/bind/aeonglobal.com** menjadi seperti ini.

```
  GNU nano 2.2.4              File: aeonglobal.com

;
; BIND data file for local loopback interface
;
$TTL    604800
@       IN      SOA     aeonglobal.com. root.aeonglobal.com. (
                              2         ; Serial
                         604800         ; Refresh
                          86400         ; Retry
                        2419200         ; Expire
                         604800 )       ; Negative Cache TTL
;
@       IN      NS      aeonglobal.com.
@       IN      A       127.0.0.1
hello   IN      CNAME   @
bye     IN      CNAME   @



                               [ Read 15 lines ]
^G Get Help  ^O WriteOut  ^R Read File ^Y Prev Page ^K Cut Text  ^C Cur Pos
^X Exit      ^J Justify   ^W Where Is  ^V Next Page ^U UnCut Text^T To Spell
```

Jangan lupa untuk merestart bind, gunakan

```
/etc/init.d/bind9 restart
```

Setelah itu, kita akan membuat sub domain di Apache agar halaman **hello.aeonglobal.com** dan **bye.aeonglobal.com** tidak sama-sama merujuk pada halaman yang sama.

Apache memiliki kemampuan untuk mendukung lebih dari satu situs. Karena itu ada direktori yang bernama **sites-available** dan **sites-enabled** di **/etc/apache2**. Masuk ke direktori **sites-available** untuk membuat sub domain **hello.aeonglobal.com**.

Gunakkan template yang sudah ada untuk domain utama, atau localhost, dengan nama **default.**

```
cp default hello.aeonglobal.com
```

Setelah itu, rubah sehingga seperti ini.

```
  GNU nano 2.2.4              File: hello.aeonglobal.com

<VirtualHost *:80>
        ServerAdmin webmaster@localhost
        ServerName hello.aeonglobal.com
        DocumentRoot /var/www/hello.aeonglobal.com
        <Directory />
                Options FollowSymLinks
                AllowOverride None
        </Directory>
        <Directory /var/www/>
                Options Indexes FollowSymLinks MultiViews
                AllowOverride None
                Order allow,deny
                allow from all
        </Directory>

        ScriptAlias /cgi-bin/ /usr/lib/cgi-bin/
        <Directory "/usr/lib/cgi-bin">
                AllowOverride None
                Options +ExecCGI -MultiViews +SymLinksIfOwnerMatch
                Order allow,deny
                              [ Wrote 31 lines ]

root@serverone:/etc/apache2/sites-available# _
```

Kita menambahkan sub domain **hello.aeonglobal.com**, di mana sumber daya yang akan disediakan berada di direktori **/var/www/hello.aeonglobal.com**.

Lakukan juga hal yang sama untuk sub domain **bye.aeonglobal.com** dengan sumber daya direktori di **/var/www/bye.aeonglobal.com**.

Setelah itu, coba buat direktori **/var/www/hello.aeonglobal.com** dan buat sebuah halaman dengan ekstensi PHP di direktori tersebut, karena kita akan mencoba menggunakan bahasa server side, yaitu php, beri nama **index.php**.

Isikan file **index.php** seperti berikut.



Lalu, lakukan hal yang sama untuk sub domain **bye.aeonglobal.com**, tapi kali ini isi dari **index.php** seperti ini.



Karena kita hanya membuat situs tersebut tersedia (berada dalam direktori **sites-available**) maka kita harus mengaktifkan situs tersebut dengan menggunakan program bawaan Apache, yaitu **a2ensite** (Apache 2 Enable Site).

Aktifkan sub domain yang sudah kita buat, gunakan

```
a2ensite <nama_direktori>
```

Dalam kasus ini seperti ini.

```
root@serverone:/usr# a2ensite hello.aeonglobal.com
Enabling site hello.aeonglobal.com.
Run '/etc/init.d/apache2 reload' to activate new configuration!
root@serverone:/usr# a2en
a2enmod   a2ensite
root@serverone:/usr# a2ensite bye.aeonglobal.com
Enabling site bye.aeonglobal.com.
Run '/etc/init.d/apache2 reload' to activate new configuration!
root@serverone:/usr# _
```

Lawan dari **a2ensite**, adalah **a2dissite** yang digunakan untuk menon-aktifkan situs yang sudah aktif.

Setelah diaktifkan, coba gunakan **wget** untuk melihat apakah hasil dari **hello.aeonglobal.com** berbeda dengan **bye.aeonglobal.com** dan juga dengan **aeonglobal.com**.

```
root@serverone:/usr# wget aeonglobal.com -O /usr/index.html
--2013-12-15 16:33:18--  http://aeonglobal.com/
Resolving aeonglobal.com... 127.0.0.1
Connecting to aeonglobal.com|127.0.0.1|:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 23 [text/html]
Saving to: "/usr/index.html"

100%[======================================>] 23          --.-K/s   in 0s

2013-12-15 16:33:18 (365 KB/s) - "/usr/index.html" saved [23/23]

root@serverone:/usr# wget hello.aeonglobal.com -O /usr/index_hello.html
--2013-12-15 16:33:27--  http://hello.aeonglobal.com/
Resolving hello.aeonglobal.com... 127.0.0.1
Connecting to hello.aeonglobal.com|127.0.0.1|:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: unspecified [text/html]
Saving to: "/usr/index_hello.html"

    [ <=>                                   ] 56,911      --.-K/s   in 0.02s

2013-12-15 16:33:27 (2.88 MB/s) - "/usr/index_hello.html" saved [56911]

root@serverone:/usr#
```

```
root@serverone:/usr# wget bye.aeonglobal.com -O /usr/index_bye.html
--2013-12-15 16:34:12--  http://bye.aeonglobal.com/
Resolving bye.aeonglobal.com... 127.0.0.1
Connecting to bye.aeonglobal.com|127.0.0.1|:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 899 [text/html]
Saving to: "/usr/index_bye.html"

100%[======================================>] 899         --.-K/s   in 0s

2013-12-15 16:34:12 (10.0 MB/s) - "/usr/index_bye.html" saved [899/899]

root@serverone:/usr# _
```

Gunakan **cat** untuk melihat isi dari masing-masing file tersebut.

index.html

```
root@serverone:/usr# cat index.html
<h1> HELLO WORLD </h1>
root@serverone:/usr# _
```

index_hello.html

```
<tr><td class="e">_SERVER["QUERY_STRING"]</td><td class="v"><i>no value</i></td>
</tr>
<tr><td class="e">_SERVER["REQUEST_URI"]</td><td class="v">/</td></tr>
<tr><td class="e">_SERVER["SCRIPT_NAME"]</td><td class="v">/index.php</td></tr>
<tr><td class="e">_SERVER["PHP_SELF"]</td><td class="v">/index.php</td></tr>
<tr><td class="e">_SERVER["REQUEST_TIME"]</td><td class="v">1387100007</td></tr>
</table><br />
<h2>PHP License</h2>
<table border="0" cellpadding="3" width="600">
<tr class="v"><td>
<p>
This program is free software; you can redistribute it and/or modify it under th
e terms of the PHP License as published by the PHP Group and included in the dis
tribution in the file:  LICENSE
</p>
<p>This program is distributed in the hope that it will be useful, but WITHOUT A
NY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR
 A PARTICULAR PURPOSE.
</p>
<p>If you did not receive a copy of the PHP license, or have any questions about
 PHP licensing, please contact license@php.net.
</p>
</td></tr>
</table><br />
</div></body></html>root@serverone:/usr# _
```

Isi dari **index_hello.html** cukup panjang, yang berisi informasi tentang modul PHP yang ada di Apache.

index_bye.html

```
    [HTTP_HOST] => bye.aeonglobal.com
    [HTTP_CONNECTION] => Keep-Alive
    [PATH] => /usr/local/bin:/usr/bin:/bin
    [SERVER_SIGNATURE] => <address>Apache/2.2.16 (Debian) Server at bye.aeonglob
al.com Port 80</address>

    [SERVER_SOFTWARE] => Apache/2.2.16 (Debian)
    [SERVER_NAME] => bye.aeonglobal.com
    [SERVER_ADDR] => 127.0.0.1
    [SERVER_PORT] => 80
    [REMOTE_ADDR] => 127.0.0.1
    [DOCUMENT_ROOT] => /var/www/bye.aeonglobal.com
    [SERVER_ADMIN] => webmaster@localhost
    [SCRIPT_FILENAME] => /var/www/bye.aeonglobal.com/index.php
    [REMOTE_PORT] => 59963
    [GATEWAY_INTERFACE] => CGI/1.1
    [SERVER_PROTOCOL] => HTTP/1.0
    [REQUEST_METHOD] => GET
    [QUERY_STRING] =>
    [REQUEST_URI] => /
    [SCRIPT_NAME] => /index.php
    [PHP_SELF] => /index.php
    [REQUEST_TIME] => 1387100052
)
root@serverone:/usr# _
```

Sekarang semua sub domain sudah berhasil terkonfigurasi dengan benar.

a. Rangkuman

HTTP server atau Web server adalah sebuah layanan yang menyediakan akses ke sumber daya dengan menggunakan protocol HTTP. Sumber daya HTTP biasanya berupa dokumen HTML, dilengkapi juga dengan dokumen beformat lainya yang mendukung HTML.

Secara default, HTTP server berjalan di atas port 80 untuk versi normal, dan berjalan di atas port 443 untuk versi aman atau HTTPS. HTTPS berjalan di atas enkripsi data sehingga data yang dikirimkan aman dan tidak bisa dicuri ditengah jalan.

HTTP server bisa bekerja secara dinamis atau statis, tergantung dari saat mereka diimplementasikan. Cara HTTP bekerja secara dinamis juga tergantung, apakah melalui software HTTP langsung atau dengan melalui bahasa pemrograman di sisi server seperti ASP atau PHP.

Ada dua jenis HTTP server, yaitu HTTP Server mode kernel, dan HTTP Server mode user. Mode kernel bisa berinteraksi dengan sumber daya tingkat rendah seperti processor, memory, adaptor jaringan. Mode kernel ini disisipkan dalam OS sehingga bekerja lebih cepat, di Windows ada IIS dan ada TUX di Linux. Mode user, berbeda jauh dan biasanya diinstall secara terpisah, karena mode user tidak mempunyai akses langsung ke sumber daya tingkat rendah, maka kecepatanya lebih rendah dan harus berbagi dengan software lainya dalam urusan sumber daya hardware.

HTTP server terbatas, dan bisa jatuh sewaktu-waktu. Karena HTTP server harus mengatasi banyak pengguna di waktu yang bersamaan, bukan tidak mungkin HTTP akan terhenti sejenak saat mengatasi banyak pengguna tersebut. Batasan-batasan ini ada sesuai dengan kualitas sistem, jenis HTTP server, perangkat keras di server, jumlah pengguna, firewall, dsb.

Beberapa cara bisa dilakukan untuk mengurangi batas HTTP server sehingga tidak sampai terhenti, bisa dilakukan dengan cara mengatur

bandwidth, firewall, menanggulangi serangan-serangan seperti DDOS, XSS, virus.

HTTP server bekerja dengan protocol HTTP, yaitu protocol yang terdiri dari dua bagian data, header dan content. Header berisi informasi tentang content yang akan disajikan, atau informasi tentang koneksi yang sedang terjadi.

Proses HTTP bekerja secara dua arah, yaitu dari pengguna ke server, lalu server ke pengguna. Ada dua jenis header yang berbentuk sama, namun dengan format dan spesifikasi yang berbeda. Setiap proses baik itu permintaan atau pelayanan selalu memenuhi format protocol HTTP.

Ada berbagai macam HTTP server yang populer, seperti Staticlet, Apache, Nginx, IIS, GlassFish, Tomcat, Node, dsb. HTTP server yang terinstal dengan sistem operasi Debian adalah Apache (apabila dipilih Web Server saat instalasi).

Apache dikenal sebagai web server paling populer dan menguasai pasar, bekerja dengan cukup, bisa dikonfigurasi sesuai kebutuhan, mendukung lebih dari satu situs, juga dikembangkan secara terus menerus oleh pembuatnya.

Instalasi Apache di Debian cukup mudah, sekali di instal dan dinyalakan, maka sudah bisa melayani permintaan http. Setelah itu Apache bisa dikonfigurasi sesuai kebutuhan untuk menambahkan situs baru, diberikan modul-modul tambahan, mengatur struktur direktori. Untungnya, Apache sudah dilengkapi dengan bahasa pemrograman sisi server yaitu PHP.



b. Tugas

1. Apa saja yang disediakan oleh web server?
2. Apa beda HTTP dan HTTPS?
3. Apa yang dimaksud dengan kernel HTTP?
4. Sebutkan batas-batas web server dan jelaskan bagaimana batas tersebut bisa tercapai!
5. Jelaskan struktur dari protokol HTTP!
6. Sebutkan jenis atau merk-merk web server minimal 5, dan jelaskan kelebihan masing!
7. Bagaimana cara konfigurasi Apache di Debian?
8. Bagaimana cara kerja web server sampai melayani pengguna?
9. Buatlah isi dari permintaan HTTP untuk pengguna yang mengakses http://aeonglobal.com menggunakan WebParticle sebagai user agent.
10. Sebutkan kelebihan Apache di banding web server lainya!

c. Test Formatif

1. Protocol apa yang digunakan oleh web server?
   a) HTTP
   b) HTTPD
   c) HTTPX
   d) HTML
   e) XHTML
2. Port mana yang digunakan oleh web server?
   a) 80,443
   b) 90,443
   c) 80,433
   d) 82,433
   e) 81,430
3. Kelebihan HTTPS adalah ...
   a) HTTPS dilengkapi dengan layanan multimedia
   b) HTTPS dilengkapi dengan keamanan yang lebih
   c) HTTPS dilengkapi dengan server
   d) HTTPS bisa digunakan untuk bermain game
   e) Semua Salah

7. File HTML tidak cukup tanpa dilengkapi media lainya, media untuk pengaturan tampilan adalah ...
    a) CCS
    b) SVG
    c) SGV
    d) CSS
    e) JS
8. Versi kernel dari HTTP mempunyai kelebihan dari HTTP versi user, yaitu ...
    a) Kernel mode dilengkapi dengan managed memory, sehingga akses bisa dibatasi
    b) Kernel mode tidak dilengkapi dengan managed memory, sehingga akses langsung ke hardware
    c) Kernel mode terpisah dari OS, sehingga bisa menghemat ruang OS
    d) Kernel mode tidak mempunyai pipe IO terbagi
    e) Semua Salah
9. Web server dibatasi oleh salah satu dari faktor berikut, kecuali ...
    a) Konfigurasi web server
    b) Jenis HTTP request
    c) Jenis konten
    d) Batas hardware atau software
    e) Semua salah
10. Serangan DDOS adalah ...
    a) Serangan berkali-kali sehingga server menolar permintaan
    b) Serangan berkali-kali sehingga server mati
    c) Serangan sekali tapi server langsung mati
    d) Serangan untuk membuat server menghapus semua file
    e) Semua salah

11. Protocol pembuka HTTP yang benar untuk pengguna yang mengakses http://aeonglobal.com/enterprise adalah ...
    a) HTTP /aeonglobal.com/enterprise GET
    b) GET /aeonglobal.com/enterprise HTTP
    c) GET /enterprise HTTP/1.1
    d) GET /enterprise HTTP
    e) GET /aeonglobal.com/enterprise HTTP/1.1
12. Gambar berikut menyatakan bahwa situs hello.aeonglobal.com mempunyai direktori yang berada di ...

```
  GNU nano 2.2.4             File: hello.aeonglobal.com

<VirtualHost *:80>
        ServerAdmin webmaster@localhost
        ServerName hello.aeonglobal.com
        DocumentRoot /var/www/hello.aeonglobal.com
        <Directory />
                Options FollowSymLinks
                AllowOverride None
        </Directory>
        <Directory /var/www/>
                Options Indexes FollowSymLinks MultiViews
                AllowOverride None
                Order allow,deny
                allow from all
        </Directory>

        ScriptAlias /cgi-bin/ /usr/lib/cgi-bin/
        <Directory "/usr/lib/cgi-bin">
                AllowOverride None
                Options +ExecCGI -MultiViews +SymLinksIfOwnerMatch
                Order allow,deny
                               [ Wrote 31 lines ]

root@serverone:/etc/apache2/sites-available# _
```

    a) /var/www/hello.aeonglobal.com
    b) /var/www
    c) /cgi-bin/usr/lib/cgi-bin
    d) /
13. Perintah 12ensite digunakan untuk ...
    a) Merubah alamat situs
    b) Mengaktifkan situs
    c) Menonaktifkan situs
    d) Menghapus situs
    e) Membuat situs

d. Lembar Jawaban Tes Formatif

e. Lembar Kerja Siswa

22. Kegiatan Belajar 2 : Menyajikan Hasil Konfigurasi Web atau HTTP Server

a. Tujuan Pembelajaran

Setelah mengikuti kegiatan belajar 2 ini, siswa diharapkan dapat:

14. Menyajikan hasil konfigurasi Web atau HTTP server dengan praktek, laporan, makalah dan presentasi.

b. Uraian Materi

15. **Penyajian Hasil Konfigurasi Web atau HTTP Server**

Siswa diberi tugas untuk membuat kelompok yang terdiri dari 3 – 5 orang, kemudian membuat sebuah web server bisa berjalan di server dengan 1 domain dan 3 sub domain.

Setelah itu siswa memberikan laporan yang berupa hasil dari praktikum, kendala yang dihadapi dan cara penyelesaianya.

Lalu siswa membuat makalah tentang penerapan web server di perusahaan besar atau dalam lingkup global, bagaimana cara mengatur sub domain yang efisien, dan melakukan konfigurasi tingkat lanjut pada Apache.

Siswa ditugaskan untuk membuat makalah menjadi lebih kompak, padat, dan singkat untuk disajikan dalam presentasi.

c. Rangkuman

d. Tugas

1. Ssiwa membuat kelompok terdiri dari 3 – 5 orang.
2. Siswa membuat sebuah web server di server dengan 1 domain dan 3 sub domain.
3. Siswa membuat laporan berdasarkan kegiatan pembuatan server, laporkan kendala dan cara menyelesaikan kendala tersebut.
4. Siswa mebuat makalah tentang implementasi web server di perusahaan besar atau dalam lingkup global, atur domain dan sub domain secare efisien. Pelajari cara konfigurasi tingkat lanjut di Apache.
5. Sajikan makalah dalam bentuk presentasi yang kompak, singkat, padat dan jelas.

e. Tes Formatif

f. Lembar Jawaban Tes Formatif

g. Lembar Kerja Siswa

Daftar Pustaka

http://www.webmonkey.com/2010/02/ftp_for_beginners/



http://www.tcpipguide.com/free/t_FTPOperationalModelProtocolComponentsandKeyTermino.htm

http://www.deskshare.com/resources/articles/ftp-how-to.aspx

http://www.windowsnetworking.com/articles-tutorials/network-protocols/Understanding-FTP-Protocol.html

http://computer.howstuffworks.com/e-mail-messaging/email1.htm

http://en.wikipedia.org/wiki/Webmail

http://en.wikipedia.org/wiki/Network_Time_Protocol

http://www.webopedia.com/TERM/R/remote_access_server.html

http://www.webopedia.com/TERM/R/remote_access_server.html

http://technet.microsoft.com/en-us/library/cc754070.aspx

http://computernetworkingnotes.com/network-administrations/telnet-server.html

http://prabulaksana.blogspot.com/2011/01/ntp-server.html

*http://lecturer.eepis-its.edu/~idris/files/admin_jarkom/***proxy***.ppt*

The TELNET Protocol, http://support.microsoft.com/kb/231866

http://searchsoa.techtarget.com/definition/

Muhammad Sasmito Adi Wibowo, 2012, instalasi dan Konfigurasi FTP Server di Debian 6, *http://cahayaspiritualitas.blogspot.com/2012/10/instalasi-dan-konfigurasi-ftp-server-di.html*

http://doenia-masguru.blogspot.com/2013/11/membuat-pc-router-dan-proxy-server.html

http://suhendartkj.blogspot.com/2013/01/cara-konfigurasi-ntp-server-di-debian-6.html

http://indrango.blogspot.com/2013/08/instalasi-dan-konfigurasi-ssh-server-di.html

